

COMPUTATION OF BALANCED EQUIVALENCE RELATIONS AND THEIR LATTICE FOR A COUPLED CELL NETWORK

HIROKO KAMEI* AND PETER J. A. COCK†

Abstract. A coupled cell network describes interacting (*coupled*) individual systems (*cells*). As in networks from real applications, coupled cell networks can represent *inhomogeneous* networks where different types of cells interact with each other in different ways, which can be represented graphically by different symbols, or abstractly by equivalence relations.

Various synchronous behaviors, from full synchrony to partial synchrony, can be observed for a given network. Patterns of synchrony, which do not depend on specific dynamics of the network, but only on the network structure, are associated with a special type of partition of cells, termed *balanced equivalence relations*. Algorithms in Aldis (2008) and Belykh and Hasler (2011) find the unique pattern of synchrony with the least clusters. In this paper, we compute the set of all possible patterns of synchrony and show their hierarchy structure as a *complete lattice*.

We represent the network structure of a given coupled cell network by a symbolic adjacency matrix encoding the different coupling types. We show that balanced equivalence relations can be determined by a matrix computation on the adjacency matrix which forms a block structure for each balanced equivalence relation. This leads to a computer algorithm to search for all possible balanced equivalence relations. Our computer program outputs the balanced equivalence relations, quotient matrices, and a complete lattice for user specified coupled cell networks. Finding the balanced equivalence relations of any network of up to 15 nodes is tractable, but for larger networks this depends on the pattern of synchrony with least clusters.

Key words. couple cell networks; synchrony; balanced equivalence relations; lattice

AMS subject classifications. 15A72, 34C14, 06B23, 90C35

1. Introduction. In many areas of science, interacting objects can be represented as a network. Examples can be found in biological, chemical, physical, technological and social systems [29, 35]. One important dynamical feature of networks is the possibility of synchrony, which occurs when distinct individuals exhibit identical dynamics. Synchronization of initially distinct dynamics, such as that appearing in fireflies, coupled lasers and coupled chaotic systems have been extensively studied (see [12, 5, 39] for reviews). Many studies have investigated the role of synchrony in a wide range of cognitive and information processing, including recently the possible relevance of neural synchrony in pathological brain states, such as epilepsy and Alzheimer’s disease (reviewed in [37]).

In this paper however we are interested in not only full synchronization, but also partial synchronization where a network breaks into sub-networks, called clusters, such that all individual systems within one cluster are perfectly synchronised. In coupled chaotic systems, such partial synchronization (or clustering) has been attracting great interest [11, 42, 10, 38]. Partial synchronies can also appear as a result of synchrony breaking. Suppose all individuals of a network are initially synchronized, but at some point lose coherence – giving synchronized sub-networks or even differently behaving individuals. Differentiation of (biological) cells [25], speciation [32], desynchronization of coupled oscillators [26], or the loss of coherence in lasers can be interpreted in this way. We are interested in computing those potential partial synchronies which are solely determined by the network architecture (topology), rather than any specific details of the network dynamics (e.g. parameter values or function forms).

*Division of Mathematics, University of Dundee, Dundee, DD1 4HN, UK (hiroko@maths.dundee.ac.uk).

†James Hutton Institute, Dundee, DD2 5DA, UK (Peter.Cock@hutton.ac.uk)

Mathematically such interaction networks are described as a directed graph [36, 41]. Nodes correspond to the individuals, and arrows (edges) denote their interactions. Coupled cell networks are a general formalism using a directed graph to describe such interacting individuals (see [34, 19, 18]). In this settings cells correspond to the individuals (graph nodes), and there can be multiple types of cell, and similarly multiple types of arrow (graph edges). The topology of the network is described by an adjacency matrix, using symbolic entries for different arrow types.

In this paper we describe how to compute all possible partial synchronies which are a consequence of the topology (i.e. the adjacency matrix for the graph) of a given coupled cell network. Such partial synchronies are represented as a partition of the network cells, termed a *balanced equivalence relation* (also referred to as a balanced coloring). These impose a block structure on the adjacency matrix, leading to a computer algorithm which determines all possible balanced equivalence relations using matrix computations. The set of all possible balanced equivalence relations is partially ordered, and forms a complete lattice (see [33]). In this paper, we compute all possible balanced equivalence relations and their complete lattice. Existing algorithms in [4, 9] can find the top lattice node, i.e. the maximal balanced equivalence relations from a given network topology. We use the top lattice node in order to reduce the search space for finding all possible balanced equivalence relations, and so speed up constructing the complete lattice.

The supplementary material includes an implementation of the algorithms described, and a hybrid of the top lattice node algorithms from [4, 9], using the freely available programming language Python (<http://www.python.org>) and Numerical Python library (<http://numpy.scipy.org>) for matrix support. We hope that interested researchers will be able to take this script and run it on networks of interest by entering the adjacency matrices. The code prints out balanced equivalence relations, their quotient network adjacency matrices, and the associated lattice structure (as text). Additionally, provided GraphViz [15] and associated Python libraries for calling it are installed, figures of the network and lattice are also produced.

Our implementation effectively finds all the balanced equivalence relations for any coupled cell network of up to 15 cells. For larger networks, the speed of computation depends on the total number of possible partitions to check, which depends on the clustering pattern of the maximal balanced equivalence relation, but not directly the size of the network. Inhomogeneous networks are most tractable, and an example of a 30 cell network is discussed.

This paper is organized as follows. In Section 2, we recall some basic features of the coupled cell formalism, then review the basics of lattice theory. In Section 3, we show that finding a balanced equivalence relation is equivalent to finding a particular type of invariant subspace of a linear map, which is represented by the adjacency matrix of a given coupled cell network. We then show that an adjacency matrix, which leaves such subspaces invariant, has a block structure. This matrix property leads to the computer algorithm discussed in Section 4 which determines all possible balanced equivalence relations of a given network, allowing display of the corresponding lattice of the set of all balanced equivalence relations. Finally in Section 5, we demonstrate how the algorithm can be applied to several examples from the literature, with conclusions in Section 6.

2. Preliminaries.

2.1. Coupled cell network and associated coupled cell system. A *coupled cell network* describes interacting individual systems schematically by a finite directed

graph $\mathcal{G} = (\mathcal{C}, \mathcal{E}, \sim_C, \sim_E)$. Here $\mathcal{C} = \{c_1, c_2, \dots, c_n\}$ is the set of nodes (*cells*), $\mathcal{E} = \{e_1, e_2, \dots, e_m\}$ is the arrows between them (*couplings*), and equivalence relations \sim_C and \sim_E describe different types of cells and couplings.

Different cell types (equivalence relation \sim_C) can be *labelled* with symbols such as circles, squares, or triangles. Similarly different arrow types (\sim_E) can be shown using different kinds of lines (solid, dashed, dotted).

Each cell c is a dynamical system with variables x_c in *cell phase space* P_c , for simplicity a finite-dimensional real vector space \mathbb{R}^r , where r may depend on c . Cells of the same type must have the same phase space, for $c, d \in \mathcal{C}$, $c \sim_C d \Rightarrow P_c = P_d$.

Each arrow $e \in \mathcal{E}$ connects a tail node $\mathcal{T}(e)$ to a head node $\mathcal{H}(e)$, expressed using maps $\mathcal{H} : \mathcal{E} \rightarrow \mathcal{C}$ and $\mathcal{T} : \mathcal{E} \rightarrow \mathcal{C}$. Arrows of the same type must have matching tail and head cell types, $e_1, e_2 \in \mathcal{E}$, $e_1 \sim_E e_2 \Rightarrow \mathcal{H}(e_1) \sim_C \mathcal{H}(e_2)$ and $\mathcal{T}(e_1) \sim_C \mathcal{T}(e_2)$, ensuring \sim_E maintains similar input/output characteristics.

For each cell $c \in \mathcal{C}$ the *input set* of c is defined as $I(c) = \{e \in \mathcal{E} : \mathcal{H}(e) = c\}$, where $e \in I(c)$ is called an *input arrow* of c .

DEFINITION 2.1. *The relation \sim_I of input equivalence on \mathcal{C} is defined by $c \sim_I d$ if and only if there exists an arrow-type preserving bijection $\beta : I(c) \rightarrow I(d)$ such that $i \sim_E \beta(i) \forall i \in I(c)$, and two cells c and d are said to be input isomorphic.*

The *input equivalence* on \mathcal{C} identifies equivalent cells whose input couplings are also equivalent. As a consequence, if $c \sim_I d$ then they should have similar dynamics defined in a *coupled cell system*, which is a system of ordinary differential equations (ODEs) associated with a given coupled cell network. Define the *total phase space* to a given n -cell coupled cell network \mathcal{G} to be $P = P_{c_1} \times \dots \times P_{c_n}$ and employ the coordinate system $(x_1, \dots, x_n) \in P$, where $x_i \in P_{c_i}$. The system associated with the cell c_i has the form

$$\dot{x}_i = f_i(x_i, x_{j_1}, \dots, x_{j_q}),$$

where the first variable of f_i represents the internal dynamics of the cell c_i and the remaining q variables $\{x_{j_1}, \dots, x_{j_q}\} = \mathcal{T}(I(c_i))$ represent coupling. In this paper, we employ definitions which permit multiple arrows (some subsets of indices j_k are equal) and self-coupling (some j_k equal i).

The function f_i corresponds to the i -th component of an *admissible vector field* $F = (f_1, \dots, f_n)$, which is compatible with the network structure, and depends on a fixed choice of the total phase space P . It follows that different components of F are identical if the corresponding cells are input isomorphic, i.e., $f_c = f_d$ for $c \sim_I d$. As a consequence, the number of distinct functions in a coupled cell system corresponds to the number of input equivalence classes. We now define types of coupled cell networks.

DEFINITION 2.2. *A homogeneous network is a coupled cell network such that all cells are input isomorphic or identical. If a coupled cell network is not homogeneous, we call it an inhomogeneous network. A homogeneous network that has one equivalence class of arrows is said to be regular. The valency of a homogeneous network is the number of arrows into each cell.*

Table 2.1 shows various types of coupled cell networks and the corresponding coupled cell systems. Note that, by definition, $c \sim_I d \Rightarrow c \sim_C d$ holds, but the converse fails in general. If $I(c) = \emptyset$ and $I(d) = \emptyset$, we say that $c \sim_I d$ iff $c \sim_C d$ (see for example the network \mathcal{G}_1 in Table 2.1).

2.2. Balanced equivalence relations and quotient networks. We say a given coupled cell system has synchrony if (at least) two cells c and d have identical

Inhomogeneous networks			
\mathcal{G}_1		$\dot{x}_1 = f_1(x_1)$ $\dot{x}_2 = g_1(x_2, \overline{x_1, x_4})$ $\dot{x}_3 = h_1(x_3, x_2, \overline{x_1, x_4})$ $\dot{x}_4 = f_1(x_4)$	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ e_1 & 0 & 0 & e_1 \\ e_2 & e_3 & 0 & e_2 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
\mathcal{G}_2		$\dot{x}_1 = f_2(x_1, x_3, x_4)$ $\dot{x}_2 = g_2(x_2, x_1)$ $\dot{x}_3 = g_2(x_3, x_1)$ $\dot{x}_4 = h_2(x_4, \overline{x_2, x_3})$	$\begin{pmatrix} 0 & 0 & e_3 & e_2 \\ e_1 & 0 & 0 & 0 \\ e_1 & 0 & 0 & 0 \\ 0 & e_4 & e_4 & 0 \end{pmatrix}$
\mathcal{G}_3		$\dot{x}_1 = f_3(x_1, x_4)$ $\dot{x}_2 = g_3(x_2, x_1)$ $\dot{x}_3 = f_3(x_3, x_2)$ $\dot{x}_4 = g_3(x_4, x_3)$ $\dot{x}_5 = g_3(x_5, x_1)$	$\begin{pmatrix} 0 & 0 & 0 & e_2 & 0 \\ e_1 & 0 & 0 & 0 & 0 \\ 0 & e_2 & 0 & 0 & 0 \\ 0 & 0 & e_1 & 0 & 0 \\ e_1 & 0 & 0 & 0 & 0 \end{pmatrix}$
Homogeneous networks			
\mathcal{G}_4		$\dot{x}_1 = f_4(x_1, x_2, x_3, x_4)$ $\dot{x}_2 = f_4(x_2, x_1, x_3, x_4)$ $\dot{x}_3 = f_4(x_3, x_2, x_1, x_4)$ $\dot{x}_4 = f_4(x_4, x_2, x_3, x_1)$	$\begin{pmatrix} 0 & e_1 & e_3 & e_2 \\ e_2 & 0 & e_1 & e_3 \\ e_3 & e_2 & 0 & e_1 \\ e_1 & e_3 & e_2 & 0 \end{pmatrix}$
\mathcal{G}_5		$\dot{x}_1 = f_5(x_1, x_2, \overline{x_2, x_4})$ $\dot{x}_2 = f_5(x_2, x_4, \overline{x_1, x_4})$ $\dot{x}_3 = f_5(x_3, x_1, \overline{x_3, x_5})$ $\dot{x}_4 = f_5(x_4, x_1, \overline{x_1, x_2})$ $\dot{x}_5 = f_5(x_5, x_1, \overline{x_3, x_5})$	$\begin{pmatrix} 0 & e_1 + e_2 & 0 & e_1 & 0 \\ e_1 & 0 & 0 & e_1 + e_2 & 0 \\ e_2 & 0 & e_1 & 0 & e_1 \\ e_1 + e_2 & e_1 & 0 & 0 & 0 \\ e_2 & 0 & e_1 & 0 & e_1 \end{pmatrix}$
Regular networks			
\mathcal{G}_6		$\dot{x}_1 = f_6(x_1, \overline{x_4, x_5})$ $\dot{x}_2 = f_6(x_2, \overline{x_1, x_4})$ $\dot{x}_3 = f_6(x_3, \overline{x_1, x_5})$ $\dot{x}_4 = f_6(x_4, \overline{x_1, x_2})$ $\dot{x}_5 = f_6(x_5, \overline{x_1, x_3})$	$e_1 \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{pmatrix}$
\mathcal{G}_7		$\dot{x}_1 = f_7(x_1, x_4)$ $\dot{x}_2 = f_7(x_2, x_1)$ $\dot{x}_3 = f_7(x_3, x_2)$ $\dot{x}_4 = f_7(x_4, x_3)$ $\dot{x}_5 = f_7(x_5, x_1)$	$e_1 \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$

TABLE 2.1

Coupled cell networks, and the corresponding coupled cell systems and symbolic adjacency matrices (defined in Section 3). The overline indicates that some couplings from other cells to that cell are identical, i.e., $f(x_i, x_j, \overline{x_k, x_l})$ means $f(x_i, x_j, x_k, x_l) = f(x_i, x_j, x_l, x_k)$.

outputs, that is $x_c(t) = x_d(t) \forall t \in \mathbb{R}$. Synchronous network dynamics solely determined by the network structure is associated with a special type of partitions of cells termed *balanced equivalence relations*, which define a smaller network called the *quotient network* describing synchronous dynamics of the original network.

Let \bowtie be an equivalence relation on \mathcal{C} , partitioning the cells into equivalence classes. For a given equivalence relation \bowtie , the corresponding subspace of the total phase space P is defined by $\Delta_{\bowtie} = \{x \in P : c \bowtie d \Rightarrow x_c = x_d\}$, which is called a *polydiagonal subspace* of P .

Denote by $\mathcal{F}_{\mathcal{G}}^P$ the class of admissible vector fields of a given coupled cell network \mathcal{G} with the total phase space P . A polydiagonal subspace is called a *synchrony subspace* (or *balanced polydiagonal*) if it is flow-invariant for every admissible vector field with the given network architecture. That is,

$$F(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie} \quad \forall F \in \mathcal{F}_{\mathcal{G}}^P.$$

Equivalently, if $x(t)$ is a trajectory of any $F \in \mathcal{F}_{\mathcal{G}}^P$, with initial condition $x(0) \in \Delta_{\bowtie}$, then $x(t) \in \Delta_{\bowtie}$ for all $t \in \mathbb{R}$. Patterns of such *robust synchrony* are classified by a special type of equivalence relation defined in the following.

DEFINITION 2.3. *An equivalence relation on \mathcal{C} is balanced if for every $c, d \in \mathcal{C}$ with $c \bowtie d$, there exists an input isomorphism β such that $\mathcal{T}(i) \bowtie \mathcal{T}(\beta(i))$ for all $i \in I(c)$, where the map $\mathcal{T}(e)$ returns the tail node of an arrow $e \in \mathcal{E}$.*

In particular, the existence of β implies $c \sim_I d$. Hence, balanced equivalence relations can only occur between input isomorphic cells. A necessary and sufficient condition for a polydiagonal subspace to be a synchrony subspace is given by:

THEOREM 2.4. *An equivalence relation \bowtie on \mathcal{C} satisfies $F(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie}$ for any admissible vector field F if and only if \bowtie is balanced.*

Proof. See [19, Theorem 4.3]. \square

A balanced equivalence relation \bowtie on a network \mathcal{G} induces a unique canonical coupled cell network $\mathcal{G}/_{\bowtie}$ on Δ_{\bowtie} , called the *quotient network*. The set of cells of the quotient network $\mathcal{G}/_{\bowtie}$ is defined as $\mathcal{C}_{\bowtie} = \{\bar{c} : c \in \mathcal{C}\}$, where \bar{c} denote the \bowtie -equivalence class of $c \in \mathcal{C}$, and the set of arrows is defined as $\mathcal{E}_{\bowtie} = \bigcup_{c \in \mathcal{S}} I(c)$, where \mathcal{S} is a set of cells consisting of precisely one cell c from each \bowtie -equivalence class.

For example, Figure 2.1 shows all balanced equivalence relations of the inhomogeneous network \mathcal{G}_3 in Table 2.1 and the corresponding quotient networks. Any dynamics on the quotient *lifts* to a synchronous dynamic on the original network \mathcal{G} .

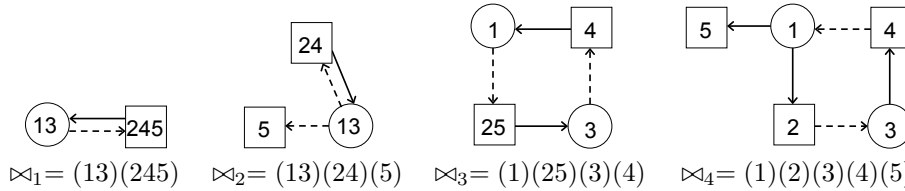


FIG. 2.1. All balanced equivalence relations of the inhomogeneous network \mathcal{G}_3 (see Table 2.1) and the corresponding quotient networks. Note that the balanced equivalence relation $\bowtie_4 = (1)(2)(3)(4)(5)$ is trivial and this gives the original 5-cell network.

2.3. Lattice theory. All possible partial synchronies (balanced equivalence relations) have a hierarchy structure represented as a *complete lattice*. We recall some basic facts about lattice theory using balanced equivalence relations as an example for some concepts. See [14] for concepts in general and for more details.

The set of all balanced equivalence relations has a partially ordered structure, using the relation of refinement. Let \bowtie_i and \bowtie_j be balanced equivalence relations on the set \mathcal{C} . Recall that \bowtie_i refines \bowtie_j , denoted by $\bowtie_i \prec \bowtie_j$, if and only if $c \bowtie_i d \Rightarrow c \bowtie_j d$ where $c, d \in \mathcal{C}$. That is, $[c]_i \subseteq [c]_j$ where $[c]_k$ is the \bowtie_k -equivalence class. The set of all balanced equivalence relations of a (locally finite) network form a complete lattice in general [33, 4].

A complete lattice has a *top* (maximal) element, denoted \top , and *bottom* (minimal) element, denoted \perp . For example, the top element of the complete lattice of balanced equivalence relations for any n -cell homogeneous network is trivial and given by $\bowtie_\top = (12 \cdots n)$ (i.e., all cells are synchronous). Aldis [4] and Belykh and Hasler [9] give algorithms to find a nontrivial maximal balanced equivalence relation (top). For any n -cell coupled cell network, the bottom element is $\bowtie_\perp = (1)(2) \cdots (n)$ (i.e., all cells are distinct).

The structure of a lattice can be visualised by a *diagram*. Let \bowtie_i, \bowtie_j and \bowtie_k be distinct balanced equivalence relations. We say \bowtie_i is covered by \bowtie_j , denoted $\bowtie_i < \bowtie_j$, if and only if $\bowtie_i \prec \bowtie_j$ and $\bowtie_i \prec \bowtie_k \prec \bowtie_j$ holds for no \bowtie_k . In a diagram, circles represent elements of the ordered set, and two elements \bowtie_i, \bowtie_j are connected by a straight line if and only if one covers the other: if \bowtie_i is covered by \bowtie_j , then the circle representing \bowtie_j is higher than the circle representing \bowtie_i . The *rank* of an equivalence relation is the number of its equivalence classes (see [23]). Figure 2.2 shows the complete lattice of the partially ordered set of all 4 balanced equivalence relations of the coupled cell network \mathcal{G}_3 , which are listed in Figure 2.1.

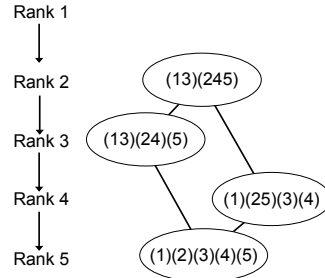


FIG. 2.2. The complete lattice of the partially ordered set of all possible balanced equivalence relations of the coupled cell network \mathcal{G}_3 along with their ranks (see also Figure 2.1). For example, the balanced equivalence relation $\bowtie_3 = (1)(25)(3)(4)$ has 4 equivalence classes, hence its rank is 4. Since this network is inhomogeneous, the top element is not of rank 1, which requires all cells are identical.

3. Matrix computation for balanced equivalence relations. We aim to determine robust patterns of synchrony of a coupled cell network solely from the network structure, which is described by the corresponding symbolic adjacency matrix defined as follows.

DEFINITION 3.1. Let $\mathcal{G} = (\mathcal{C}, \mathcal{E}, \sim_C, \sim_E)$ be an n -cell coupled cell network with l cell-types and m arrow-types with $[c_1]_C, \dots, [c_l]_C$, the \sim_C -equivalence classes for cells and $[e_1]_E, \dots, [e_m]_E$, the \sim_E -equivalence classes for arrows. We define the symbolic adjacency matrix of \mathcal{G} to be the $n \times n$ matrix $A = (a_{ij})$. The (i, j) -entry corresponds to the number of arrows of types $[e_1]_E, \dots, [e_m]_E$ from cell j to cell i , represented by the sum $\sum_{k=1}^m \beta_k e_k$, where e_k is the type of arrow corresponding to the $[e_k]_E$ -equivalence class and β_k is the number of arrows of type e_k .

Example coupled cell networks and their corresponding symbolic adjacency matrices are shown in Table 2.1.

Our main result determines all possible balanced equivalence relations combinatorially using matrix manipulations. In Lemma 3.8, we show that the necessary and sufficient condition for a synchrony subspace imposes a matrix property defined as follows.

DEFINITION 3.2. *Let $B = (b_{ij})$ be a $p \times q$ symbolic matrix. We say B is a homogenous block matrix if the sum $\sum_{j=1}^q b_{ij}$ is identical for all rows $i = 1, \dots, p$.*

The polydiagonal Δ_{\bowtie} is defined by a given equivalence relation \bowtie which determines a unique partition of cells. We use normal form cycle notation which is obtained by writing the cell numbers $1, \dots, n$ in increasing order in each cycle, starting with the 1-cycle, then the 2-cycles, and so on in increasing order of length. For example, the following polydiagonal subspace $\Delta_{\bowtie} = \{(x_1, x_2, x_3, x_4, x_5, x_6) | x_2 = x_4, x_3 = x_5 = x_6\}$ corresponds to the equivalence relation $\bowtie = (1)(24)(356)$ and can be written as $\bowtie = [1^1 2^1 3^1]$. Let $A = (a_{ij})$, $i, j = 1, \dots, 6$ be the adjacency matrix of a 6-cell coupled cell network. For the above normal form cycle notation of the equivalence relation \bowtie , we arrange the columns and rows of the adjacency matrix of the network accordingly as follows:

$$A = \left(\begin{array}{c|c|c|c|c|c} a_{11} & a_{12} & a_{14} & a_{13} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{24} & a_{23} & a_{25} & a_{26} \\ a_{41} & a_{42} & a_{44} & a_{43} & a_{45} & a_{46} \\ \hline a_{31} & a_{32} & a_{34} & a_{33} & a_{35} & a_{36} \\ a_{51} & a_{52} & a_{54} & a_{53} & a_{55} & a_{56} \\ a_{61} & a_{62} & a_{64} & a_{63} & a_{65} & a_{66} \end{array} \right) = \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix}$$

This is a block matrix, with 3×3 blocks, where 3 is the number of equivalence classes.

More generally, let $\bowtie = [1^{\alpha_1} 2^{\alpha_2} \dots n^{\alpha_n}]$ and $k = \sum_{i=1}^n \alpha_i$ be the number of equivalence classes of \bowtie which determines the polydiagonal Δ_{\bowtie} . Interchanging the adjacency matrix rows and columns to match the permutation normal form gives a block matrix with $k \times k$ blocks.

Our main result (Theorem 3.9) states that a polydiagonal subspace Δ_{\bowtie} is a synchrony subspace if and only if each block of the (reordered) adjacency matrix A is a homogeneous block matrix.

Alternatively, our result can be obtained by defining one (integer entry) adjacency matrix per arrow type (as in [3] for regular networks with one arrow type), and finding the intersection of balanced equivalence relations for the arrow-specific adjacency matrices.

3.1. Linear admissible vector fields. We represent the matrix form of \mathcal{G} -admissible linear vector fields on an n -cell coupled cell network \mathcal{G} .

DEFINITION 3.3. *Let $S = (s_{ij})$ and $T = (t_{ij})$ be symbolic matrices with the same size. Suppose $s_{ij} = s_{i'j'}$ for some (i, j) -th and (i', j') -th entries of S . If the corresponding entries in T satisfy $t_{ij} = t_{i'j'}$ for all such indices (i, j) and (i', j') , then we denote this relation by $S \sim T$.*

PROPOSITION 3.4. *Let $A = (a_{ij})$ be the $n \times n$ symbolic adjacency matrix of an n -cell coupled cell network \mathcal{G} . The $n \times n$ symbolic matrix $J = (J_{ij})$ of \mathcal{G} -admissible linear vector fields on \mathcal{G} has the form:*

$$J = D + \tilde{A}, \tag{3.1}$$

where $D = (d_{ij})$ is an $n \times n$ diagonal matrix with $d_{ii} = d_{kk}$ when $i \sim_I k$ for $i, k \in \mathcal{C}$, and $\tilde{A} = (\tilde{a}_{ij})$ is an $n \times n$ matrix such that $A \sim \tilde{A}$.

Proof. Let r_i be the dimension of internal dynamics of the i -th cell. Let \bar{J} be the $n' \times n'$ matrix form of the \mathcal{G} -admissible linear vector field of the network \mathcal{G} with total phase space $P' = \mathbb{R}^{n'}$ where $n' = \sum_{i=1}^n r_i$.

\bar{J} can be described as a block matrix $\bar{J} = (\bar{J}_{ij})$, $i, j = 1, \dots, n$, and each block of \bar{J}_{ij} is a $k_i \times k_j$ matrix with real entries. We can decompose \bar{J} as

$$\bar{J} = \bar{D} + (-\bar{D} + \bar{J}).$$

Each block of the diagonal matrix \bar{D}_{ii} satisfies $D_{ii} = D_{kk}$ when $i \sim_I k$ for $i, k \in \mathcal{C}$. By representing each block \bar{D}_{ii} with a symbol d_{ii} , we obtain an $n \times n$ symbolic diagonal matrix D . Similarly, by representing each block \bar{J}_{ij} of $-\bar{D} + \bar{J}$ with a symbol \tilde{a}_{ij} , we obtain an $n \times n$ matrix \tilde{A} , which satisfies $A \sim \tilde{A}$ by the admissibility. Therefore, the $n \times n$ symbolic matrix $J = (J_{ij})$ of \mathcal{G} -admissible linear vector fields on an n -cell coupled cell network \mathcal{G} has the form $J = D + \tilde{A}$. \square

An equivalence relation \bowtie is balanced if and only if the \mathcal{G} -admissible linear vector field J satisfies $J(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie}$ [19, Theorem 4.3]. This fact leads to a generalization of Proposition 4.1. in [23], which consists of only one type of cell and one type of coupling.

PROPOSITION 3.5. *Let \mathcal{G} be a coupled cell network associated with the admissible vector field F . Let A be the adjacency matrix of \mathcal{G} . \bowtie is balanced if and only if $A(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie}$ where Δ_{\bowtie} is a synchrony subspace associated with \bowtie .*

Proof. Using Equation 3.1, the result follows immediately:

$$\begin{aligned} & J(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie} \\ \Leftrightarrow & \{D + \tilde{A}\}(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie} \\ \Leftrightarrow & \{D(\Delta_{\bowtie}) + \tilde{A}(\Delta_{\bowtie})\} \subseteq \Delta_{\bowtie} \\ \Leftrightarrow & \tilde{A}(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie} \quad \text{since } \bowtie \text{ refines } \sim_I, \text{ and therefore } D(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie} \\ \Leftrightarrow & A(\Delta_{\bowtie}) \subseteq \Delta_{\bowtie} \quad \text{since } A \sim \tilde{A}. \end{aligned}$$

\square

Now, we aim to determine all possible invariant polydiagonals under a given adjacency matrix using a projection map. Since the determination of synchrony subspaces does not depend on the size of internal dynamics of the cells, without loss of generality, we assume the total phase space $P = \mathbb{R}^n$ for an n -cell coupled cell network for the remaining arguments.

3.2. Projection onto a polydiagonal. We construct a projection map on a given polydiagonal as follows.

Let $\Delta_{\bowtie} \subseteq \mathbb{R}^n$ be a polydiagonal subspace of \mathbb{R}^n , and Δ'_{\bowtie} denote its complement. We define the projection map P_{\bowtie} of \mathbb{R}^n on Δ_{\bowtie} along Δ'_{\bowtie} by $\text{Im}(P_{\bowtie}) = \Delta_{\bowtie}$ and $\text{Ker}(P_{\bowtie}) = \Delta'_{\bowtie}$.

Let Δ_{\bowtie} be the polydiagonal determined by a given equivalence relation \bowtie . We use normal form cycle notation for an equivalence relation. For example, the equivalence relation \bowtie corresponding to a polydiagonal subspace $\Delta_{\bowtie} = \{(x_1, x_2, x_3) | x_2 = x_3\}$ is written as a product of disjoint cycles $\bowtie = (1)(23)$ in normal form, and also written as $\bowtie = [1^1 2^1]$.

Now we define a map π which maps each element to the first element of the cycle that they belong to when written in normal form. For example, the elements in the above product of disjoint cycles are mapped to $1 \rightarrow 1$, $2 \rightarrow 2$, and $3 \rightarrow 2$ by a map π . We define the corresponding projection matrix $P_{\bowtie} = (p_{ij})$ on Δ_{\bowtie} , which is written in normal form of a partition using the map π as follows:

$$p_{i,\pi(i)} = 1$$

with all other entries being 0.

With the elements of P_{\bowtie} defined in this way, the projection matrix has a block diagonal form:

$$P_{\bowtie} = \begin{pmatrix} P_1 & 0 & \cdots & 0 \\ 0 & P_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & P_k \end{pmatrix}$$

where k is the number of disjoint cycles and P_i , $i = 1, \dots, k$ is a $t_i \times t_i$ square projection matrix on $\Delta = \{(x_1, \dots, x_{t_i}) | x_1 = \dots = x_{t_i}\}$ with $\text{rank}(P_i) = 1 \ \forall i$ and off-diagonal blocks are zero matrices.

LEMMA 3.6. *Let P_{\bowtie} and A be linear mappings of \mathbb{R}^n and let $\mathbb{R}^n = \Delta_{\bowtie} \oplus \Delta'_{\bowtie}$. Δ_{\bowtie} is A -invariant if and only if $P_{\bowtie}AP_{\bowtie} = AP_{\bowtie}$, where P_{\bowtie} is the projection on Δ_{\bowtie} along Δ'_{\bowtie} and A is the adjacency matrix of a given coupled cell network.*

Proof. This result is well known, see for example [27] and [23]. \square

For the rest of arguments, we arrange the columns and rows of the adjacency matrix of the network according to the normal form of a given equivalence relation \bowtie .

PROPOSITION 3.7. *Δ_{\bowtie} is a synchrony subspace of a coupled cell network \mathcal{G} if and only if $P_{\bowtie}AP_{\bowtie} = AP_{\bowtie}$, where P_{\bowtie} is the projection on Δ_{\bowtie} along Δ'_{\bowtie} and A is the adjacency matrix of \mathcal{G} .*

Proof. If Δ_{\bowtie} is a synchrony subspace, then the corresponding equivalence relation \bowtie is balanced. Hence, by Proposition 3.5, Δ_{\bowtie} is A -invariant and from Lemma 3.6, the corresponding projection map P_{\bowtie} to Δ_{\bowtie} satisfies $P_{\bowtie}AP_{\bowtie} = AP_{\bowtie}$. Conversely, if $P_{\bowtie}AP_{\bowtie} = AP_{\bowtie}$, then the corresponding subspace Δ_{\bowtie} is A -invariant by Lemma 3.6 and Proposition 3.5, thus the corresponding polydiagonal is balanced. \square

3.3. Block structure of an adjacency matrix. We show that the necessary and sufficient condition for a synchrony subspace in Proposition 3.7 imposes a block structure on the adjacency matrix.

LEMMA 3.8. *Let A be the $n \times n$ adjacency matrix of a given coupled cell network \mathcal{G} . Suppose a synchrony subspace Δ_{\bowtie} is defined by a partition $[1^{\alpha_1} 2^{\alpha_2} \dots n^{\alpha_n}]$ and the corresponding projection matrix P_{\bowtie} is a block diagonal matrix whose diagonal blocks P_i , $i = 1, \dots, k$, where $k = \sum_{i=1}^n \alpha_i$ are projection matrices on diagonal subspaces Δ in the corresponding dimensions. Then $P_{\bowtie}AP_{\bowtie} = AP_{\bowtie}$ if and only if corresponding blocks of A to P_{\bowtie} satisfy the following condition:*

- Let A_{st} , where $s, t = 1, \dots, k$ be blocks of A . For all blocks A_{st} , the sum of each row is identical.

Proof. Since $P_{\bowtie} = \begin{pmatrix} P_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & P_k \end{pmatrix}$, we obtain:

$$\begin{aligned}
AP_{\bowtie} &= \begin{pmatrix} A_{11} & \cdots & A_{1k} \\ \vdots & \ddots & \vdots \\ A_{k1} & \cdots & A_{kk} \end{pmatrix} \begin{pmatrix} P_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & P_k \end{pmatrix} \\
&= \begin{pmatrix} A_{11}P_1 & A_{12}P_2 & \cdots & A_{1k}P_k \\ A_{21}P_1 & A_{22}P_2 & \cdots & A_{2k}P_k \\ \vdots & \vdots & \ddots & \vdots \\ A_{k1}P_1 & A_{k2}P_2 & \cdots & A_{kk}P_k \end{pmatrix} \\
P_{\bowtie}AP_{\bowtie} &= \begin{pmatrix} P_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & P_k \end{pmatrix} \begin{pmatrix} A_{11}P_1 & A_{12}P_2 & \cdots & A_{1k}P_k \\ A_{21}P_1 & A_{22}P_2 & \cdots & A_{2k}P_k \\ \vdots & \vdots & \ddots & \vdots \\ A_{k1}P_1 & A_{k2}P_2 & \cdots & A_{kk}P_k \end{pmatrix} \\
&= \begin{pmatrix} P_1A_{11}P_1 & P_1A_{12}P_2 & \cdots & P_1A_{1k}P_k \\ P_2A_{21}P_1 & P_2A_{22}P_2 & \cdots & P_2A_{2k}P_k \\ \vdots & \vdots & \ddots & \vdots \\ P_kA_{k1}P_1 & P_kA_{k2}P_2 & \cdots & P_kA_{kk}P_k \end{pmatrix}
\end{aligned}$$

Hence, $P_{\bowtie}AP_{\bowtie} = AP_{\bowtie} \iff P_sA_{st}P_t = A_{st}P_t$, for all $s, t = 1, \dots, k$.

Let $A_{st} = (a_{ij}^{st})$ be an arbitrary $l \times m$ block matrix. Then P_t is a $m \times m$ square matrix and P_s is a $l \times l$ square matrix. Since P_t and P_s are projection matrices onto m -dimensional and l -dimensional diagonals, respectively,

$$\begin{aligned}
A_{st}P_t &= \begin{pmatrix} a_{11}^{st} & \cdots & a_{1m}^{st} \\ \vdots & \ddots & \vdots \\ a_{l1}^{st} & \cdots & a_{lm}^{st} \end{pmatrix} \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix} \\
&= \begin{pmatrix} a_{11}^{st} + \cdots + a_{1m}^{st} & 0 & \cdots & 0 \\ a_{21}^{st} + \cdots + a_{2m}^{st} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1}^{st} + \cdots + a_{lm}^{st} & 0 & \cdots & 0 \end{pmatrix} \\
P_sA_{st}P_t &= \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} a_{11}^{st} + \cdots + a_{1m}^{st} & 0 & \cdots & 0 \\ a_{21}^{st} + \cdots + a_{2m}^{st} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1}^{st} + \cdots + a_{lm}^{st} & 0 & \cdots & 0 \end{pmatrix} \\
&= \begin{pmatrix} a_{11}^{st} + \cdots + a_{1m}^{st} & 0 & \cdots & 0 \\ a_{11}^{st} + \cdots + a_{1m}^{st} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{11}^{st} + \cdots + a_{1m}^{st} & 0 & \cdots & 0 \end{pmatrix}
\end{aligned}$$

Thus $P_sA_{st}P_t = A_{st}P_t$ if and only if $\sum_{j=1}^m a_{ij}^{st}$ is identical for all $i = 1, \dots, l$.

Therefore, $P_s A_{st} P_t = A_{st} P_t$ for all $s, t = 1, \dots, k$ if and only if, for all blocks A_{st} , the sum of each row is identical. \square

We call a block of this form a *homogeneous block matrix*. It now follows immediately that:

THEOREM 3.9. *A polydiagonal subspace \triangle_{\bowtie} is a synchrony subspace if and only if each block of the adjacency matrix A , which corresponds to a block of P_{\bowtie} , is a homogeneous block matrix.*

Proof. Each block of A is a homogeneous block matrix

$$\iff P_{\bowtie} A P_{\bowtie} = A P_{\bowtie} \text{ (by Lemma 3.8)}$$

$$\iff \triangle_{\bowtie} \text{ is a synchrony subspace (by Proposition 3.7). } \square$$

EXAMPLE 3.1. *The projection mapping on $\triangle_{\bowtie} = \{(x_1, x_2, x_3) | x_2 = x_3\}$ has the form:*

$$P_{\bowtie} = \left(\begin{array}{c|cc} 1 & 0 & 0 \\ \hline 0 & 1 & 0 \\ 0 & 1 & 0 \end{array} \right)$$

Let A be the adjacency matrix of a 3-cell coupled cell network of the form:

$$A = \left(\begin{array}{c|cc} a_{11} & a_{12} & a_{13} \\ \hline a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{array} \right)$$

Then,

$$\begin{aligned} P_{\bowtie} A P_{\bowtie} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} + a_{13} & 0 \\ a_{21} & a_{22} + a_{23} & 0 \\ a_{31} & a_{32} + a_{33} & 0 \end{pmatrix} \\ &= \begin{pmatrix} a_{11} & a_{12} + a_{13} & 0 \\ a_{21} & a_{22} + a_{23} & 0 \\ a_{21} & a_{22} + a_{23} & 0 \end{pmatrix} \end{aligned}$$

Therefore,

$$P_{\bowtie} A P_{\bowtie} = A P_{\bowtie} \iff a_{21} = a_{31} \quad \text{and} \quad a_{22} + a_{23} = a_{32} + a_{33}.$$

Hence, \triangle_{\bowtie} is a synchrony subspace if and only if A has the following block structure:

$$A = \left(\begin{array}{c|cc} a_{11} & a_{12} & a_{13} \\ \hline a_{21} & a_{22} & a_{23} \\ a_{21} & a_{32} & a_{33} \end{array} \right)$$

where $a_{22} + a_{23} = a_{32} + a_{33}$. \diamond

Next we derive the adjacency matrix of the quotient network which is defined as the adjacency matrix A restricted on a synchrony subspace \triangle_{\bowtie} .

COROLLARY 3.10. *Let \triangle_{\bowtie} be a synchrony subspace defined by a partition $[1^{\alpha_1} 2^{\alpha_2} \dots n^{\alpha_n}]$ with $k = \alpha_1 + \dots + \alpha_n$ equivalence classes, and P_{\bowtie} be the corresponding block projection matrix. Let A_{st} for $s, t = 1, \dots, k$ corresponding to the blocks of P_{\bowtie} be blocks of*

an $n \times n$ adjacency matrix A . If blocks $A_{st} = (a^{st})_{ij}$ are homogeneous block matrices such that:

$$A = \left(\begin{array}{c|c|c|c|c|c|c|c|c|c|c} A_{11} & \cdots & A_{1\alpha_1} & A_{1(\alpha_1+1)} & \cdots & A_{1(\alpha_1+\alpha_2)} & \cdots & A_{1(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & A_{1k} \\ \hline A_{21} & \cdots & A_{2\alpha_1} & A_{2(\alpha_1+1)} & \cdots & A_{2(\alpha_1+\alpha_2)} & \cdots & A_{2(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & A_{2k} \\ \hline \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \hline A_{k1} & \cdots & A_{k\alpha_1} & A_{k(\alpha_1+1)} & \cdots & A_{k(\alpha_1+\alpha_2)} & \cdots & A_{k(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & A_{kk} \end{array} \right)$$

then the quotient network corresponding to \bowtie has a $k \times k$ adjacency matrix $A|_{\Delta_{\bowtie}}$, denoted by A_{\bowtie} , of the form:

$$A_{\bowtie} = \left(\begin{array}{c|c|c|c|c|c|c|c|c|c|c} \sum_{j=1}^1 a_{1j}^{11} & \cdots & \sum_{j=1}^1 a_{1j}^{1\alpha_1} & \cdots & \sum_{j=1}^n a_{1j}^{1(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & \sum_{j=1}^n a_{1j}^{1k} \\ \hline \sum_{j=1}^1 a_{1j}^{21} & \cdots & \sum_{j=1}^1 a_{1j}^{2\alpha_1} & \cdots & \sum_{j=1}^n a_{1j}^{2(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & \sum_{j=1}^n a_{1j}^{2k} \\ \hline \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \hline \sum_{j=1}^1 a_{1j}^{k1} & \cdots & \sum_{j=1}^1 a_{1j}^{k\alpha_1} & \cdots & \sum_{j=1}^n a_{1j}^{k(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & \sum_{j=1}^n a_{1j}^{kk} \end{array} \right)$$

Proof. Let $\{v_1, \dots, v_{\alpha_1}, v_{\alpha_1+1}, \dots, v_k\}$ be a basis of a synchrony subspace. Each basis element corresponds to a conjugacy class of the partition \bowtie and is an $n \times 1$ vector. Therefore, the basis elements have the following forms:

$$\begin{aligned} v_1 &= [1, 0, \dots, 0]^t \\ &\vdots \\ v_{\alpha_1} &= [\underbrace{0, \dots, 0}_{\alpha_1-1}, 1, 0, \dots, 0]^t \\ v_{\alpha_1+1} &= [\underbrace{0, \dots, 0}_{\alpha_1}, \underbrace{1, 1}_2, 0, \dots, 0]^t \\ &\vdots \\ v_k &= [\underbrace{0, \dots, 0}_{\sum_{i=1}^{n-1} i\alpha_i}, \underbrace{1, \dots, 1}_n]^t \end{aligned}$$

Since each block A_{st} is a homogeneous block matrix, i.e., the sum of each row is identical, we can express the image of each basis element using a linear combination of a basis with the sum of the first row of each A_{st} being used as a coefficient such as:

$$\begin{aligned}
Av_1 &= \begin{pmatrix} \sum_{j=1}^1 a_{1j}^{11} \\ \sum_{j=1}^1 a_{1j}^{21} \\ \vdots \\ \sum_{j=1}^1 a_{1j}^{k1} \end{pmatrix} \\
&= \sum_{j=1}^1 a_{1j}^{11} v_1 + \sum_{j=1}^1 a_{1j}^{21} v_2 + \cdots + \sum_{j=1}^1 a_{1j}^{k1} v_k \\
Av_2 &= \begin{pmatrix} \sum_{j=1}^1 a_{1j}^{12} \\ \sum_{j=1}^1 a_{1j}^{22} \\ \vdots \\ \sum_{j=1}^1 a_{1j}^{k2} \end{pmatrix} \\
&= \sum_{j=1}^1 a_{1j}^{12} v_1 + \sum_{j=1}^1 a_{1j}^{22} v_2 + \cdots + \sum_{j=1}^1 a_{1j}^{k2} v_k \\
&\vdots \\
Av_k &= \begin{pmatrix} \sum_{j=1}^n a_{1j}^{1k} \\ \sum_{j=1}^n a_{1j}^{2k} \\ \vdots \\ \sum_{j=1}^n a_{1j}^{kk} \end{pmatrix} \\
&= \sum_{j=1}^n a_{1j}^{1k} v_1 + \sum_{j=1}^n a_{1j}^{2k} v_2 + \cdots + \sum_{j=1}^n a_{1j}^{kk} v_k
\end{aligned}$$

Therefore a $k \times k$ matrix A_{\bowtie} , which is the adjacency matrix A restricted on Δ_{\bowtie} , is written as

$$A_{\bowtie} = \left(\begin{array}{c|c|c|c|c|c|c} \sum_{j=1}^1 a_{1j}^{11} & \cdots & \sum_{j=1}^1 a_{1j}^{1\alpha_1} & \cdots & \sum_{j=1}^n a_{1j}^{1(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & \sum_{j=1}^n a_{1j}^{1k} \\ \sum_{j=1}^1 a_{1j}^{21} & \cdots & \sum_{j=1}^1 a_{1j}^{2\alpha_1} & \cdots & \sum_{j=1}^n a_{1j}^{2(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & \sum_{j=1}^n a_{1j}^{2k} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \sum_{j=1}^1 a_{1j}^{k1} & \cdots & \sum_{j=1}^1 a_{1j}^{k\alpha_1} & \cdots & \sum_{j=1}^n a_{1j}^{k(\alpha_1+\cdots+\alpha_{n-1}+1)} & \cdots & \sum_{j=1}^n a_{1j}^{kk} \end{array} \right)$$

and this is the adjacency matrix of the quotient network corresponding to \bowtie . \square

4. Computer algorithms.

4.1. Balanced equivalence relations. The above combinatorial properties of adjacency matrices leads to a computer algorithm which determines all balanced equivalence relations and adjacency matrices A_{\bowtie} of associated quotient networks $\mathcal{G}/_{\bowtie}$ for a given coupled network \mathcal{G} .

We enumerate all possible equivalence relations \bowtie of n -cells, and test which are balanced. If the top lattice node is known in advance, e.g. using the algorithm in [4] or [9], only equivalence relations which refine this top node need be tested. To test if each \bowtie is balanced, we construct an $n \times k$ matrix, where k is the number

of equivalence classes of \bowtie , from the $n \times n$ adjacency matrix of \mathcal{G} . All rows in each equivalence class are identical for a balanced equivalence relation. Finally, for balanced equivalence relations, we can construct adjacency matrices A_{\bowtie} of the corresponding quotient networks $\mathcal{G}/_{\bowtie}$.

Step 1: For a given n -cell coupled cell network $\mathcal{G} = (\mathcal{C}, \mathcal{E}, \sim_C, \sim_E)$, we express the corresponding $n \times n$ adjacency matrix A as

$$A = [C_1 \cdots C_n]$$

where $C_i \in \mathbb{R}^{n \times 1}, i = 1, \dots, n$ are column vectors. Let \overline{C}_p denote the \bowtie -equivalence classes on \mathcal{C} where $p = 1, \dots, k$. For example, if $\mathcal{C} = \{\{1, 3, 5\}, \{2\}, \{4\}\}$ then $\overline{C}_1 = \{1, 3, 5\}$, $\overline{C}_2 = \{2\}$, and $\overline{C}_3 = \{4\}$. Note that $\sum_{p=1}^k |\overline{C}_p| = n$. Let \overline{C}_{p1} be the first element of each equivalence class. We assume that $\overline{C}_{11} < \overline{C}_{21} < \cdots < \overline{C}_{k1}$, where these cell numbers are used as indices for row vectors in Step 3.

We generate a new $n \times k$ matrix \tilde{A}_{\bowtie} with columns

$$\tilde{C}_{\bowtie p} = \sum_{j \in \overline{C}_p} C_j \quad \text{for } p = 1, \dots, k$$

for all possible equivalence relations \bowtie .

Let $\tilde{R}_{\bowtie i} \in \mathbb{R}^{1 \times k}$, where $i = 1, \dots, n$, denote the row vectors of this new $n \times k$ matrix \tilde{A}_{\bowtie} . Therefore,

$$\tilde{A}_{\bowtie} = \begin{pmatrix} \tilde{R}_{\bowtie 1} \\ \vdots \\ \tilde{R}_{\bowtie n} \end{pmatrix} = [\tilde{C}_{\bowtie 1} \cdots \tilde{C}_{\bowtie k}]$$

Step 2: Now we determine which equivalence relations \bowtie are balanced. An equivalence relation \bowtie on \mathcal{C} is balanced if and only if for all $p = 1, \dots, k$ we have:

$$\tilde{R}_{\bowtie l} = \tilde{R}_{\bowtie m} \quad \forall l, m \in \overline{C}_p \quad (4.1)$$

Step 3: If the above condition 4.1 is satisfied, the $k \times k$ adjacency matrix of the quotient network A_{\bowtie} corresponding to a balanced equivalence relation \bowtie is given by:

$$A_{\bowtie} = \begin{pmatrix} R_{\bowtie 1} \\ \vdots \\ R_{\bowtie k} \end{pmatrix}$$

where $R_{\bowtie i} \in \mathbb{R}^{1 \times k}$, $i = 1, \dots, k$ are representative row vectors in each equivalence class.

EXAMPLE 4.1. Consider the homogeneous network \mathcal{G}_5 in Table 2.1 with the corresponding adjacency matrix shown in Figure 4.1.

We determine if the equivalence relation (i.e., partition of cells) $\bowtie = (135)(24)$ is balanced or not by the matrix computation. There are two equivalence classes $\overline{C}_1 = \{1, 3, 5\}$ and $\overline{C}_2 = \{2, 4\}$. We generate a new 5×2 matrix \tilde{A}_{\bowtie} by adding columns 1, 3, 5 and 2, 4 such that

$$\tilde{A}_{\bowtie} = \begin{pmatrix} 0 & 2e_1 + e_2 \\ e_1 & e_1 + e_2 \\ 2e_1 + e_2 & 0 \\ e_1 + e_2 & e_1 \\ 2e_1 + e_2 & 0 \end{pmatrix}$$

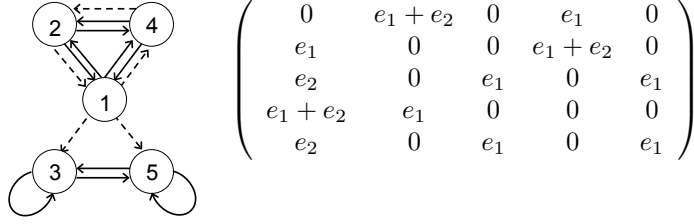


FIG. 4.1. Homogeneous network \mathcal{G}_5 in Table 2.1 with the corresponding symbolic adjacency matrix.

The equivalence relation \bowtie is balanced if and only if

$$[0 \quad 2e_1 + e_2] = [2e_1 + e_2 \quad 0] = [2e_1 + e_2 \quad 0] \quad \text{and} \quad [e_1 \quad e_1 + e_2] = [e_1 + e_2 \quad e_1].$$

However, this does not hold. Thus the equivalence relation $\bowtie = (135)(24)$ is not balanced.

On the other hand, let $\bowtie = (124)(3)(5)$. There are three equivalence classes $\overline{C}_1 = \{1, 2, 4\}$, $\overline{C}_2 = \{3\}$ and $\overline{C}_3 = \{5\}$. We generate a new 5×3 matrix \tilde{A}_{\bowtie} by adding columns 1, 2, 4 such that

$$\tilde{A}_{\bowtie} = \begin{pmatrix} 2e_1 + e_2 & 0 & 0 \\ 2e_1 + e_2 & 0 & 0 \\ e_2 & e_1 & e_1 \\ 2e_1 + e_2 & 0 & 0 \\ e_2 & e_1 & e_1 \end{pmatrix}$$

The equivalence relation \bowtie is balanced if and only if

$$[2e_1 + e_2 \quad 0 \quad 0] = [2e_1 + e_2 \quad 0 \quad 0] = [2e_1 + e_2 \quad 0 \quad 0].$$

This is satisfied. Thus the equivalence relation $\bowtie = (124)(3)(5)$ is balanced. As a result, the quotient network \mathcal{G}/\bowtie corresponding to the balanced equivalence relation $\bowtie = (124)(3)(5)$ and the associated 3×3 adjacency matrix A_{\bowtie} are given in Figure 4.2.

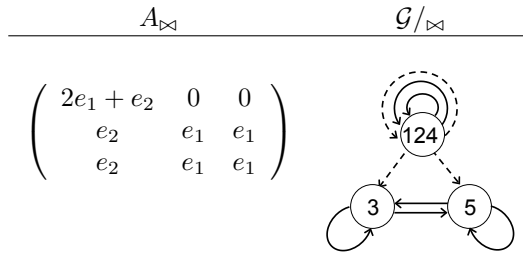


FIG. 4.2. The quotient network \mathcal{G}/\bowtie corresponding to a balanced equivalence relation $\bowtie = (124)(3)(5)$ and the associated 3×3 adjacency matrix A_{\bowtie} .

The algorithm as shown above describes the graph \mathcal{G} with a single adjacency matrix A containing symbolic entries for the different arrow types. We now discuss an alternative representation using separate integer matrices for each arrow type, which for most programming languages is more practical to implement. The following definition is a variation of Definition 5.2 in [2] for homogeneous networks.

DEFINITION 4.1. Let $\mathcal{G} = (\mathcal{C}, \mathcal{E}, \sim_C, \sim_E)$ be an n -cell coupled cell network with l cell-types and m arrow-types with $[c_1]_C, \dots, [c_l]_C$, the \sim_C -equivalence classes for cells

and $[e_1]_E, \dots, [e_m]_E$, the \sim_E -equivalence classes for arrows. We define the adjacency matrix of \mathcal{G} with respect to $[e_k]_E$, for $k = 1, \dots, m$ to be the $n \times n$ matrix $M_{(\mathcal{G},k)}$. The (i,j) -entry corresponds to the number of arrows of types $[e_k]_E$ from cell j to cell i .

Notice by construction we have:

$$A = \sum_{k=1}^m e_k M_{(\mathcal{G},k)}.$$

Therefore the above algorithm procedure can now be applied to each of the m arrow type specific matrices individually, and if it holds for all of them, it also holds for A as well. We now repeat Example 4.1 to demonstrate this.

EXAMPLE 4.2. For the coupled cell network \mathcal{G} , two adjacency matrices $M_{(\mathcal{G},1)} = (m_{ij}^1)$ (solid) and $M_{(\mathcal{G},2)} = (m_{ij}^2)$ (dashed) for two different arrow types are defined as in Figure 4.3.

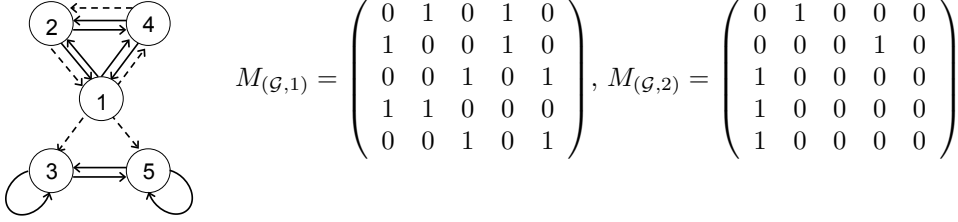


FIG. 4.3. Homogeneous network \mathcal{G}_5 in Table 2.1 with two adjacency matrices $M_{(\mathcal{G},1)} = (m_{ij}^1)$ (solid) and $M_{(\mathcal{G},2)} = (m_{ij}^2)$ (dashed) for two different arrow types.

Using these arrow type specific adjacency matrices, we determine if the equivalence relation $\bowtie = (135)(24)$ is balanced. There are two equivalence classes $\overline{C}_1 = \{1, 3, 5\}$ and $\overline{C}_2 = \{2, 4\}$. We generate new 5×2 matrices by adding vectors in columns 1, 3, 5 and 2, 4:

$$\widetilde{M}_{(\mathcal{G},1)}_{\bowtie} = \begin{pmatrix} 0 & 2 \\ 1 & 1 \\ 2 & 0 \\ 1 & 1 \\ 2 & 0 \end{pmatrix}, \quad \widetilde{M}_{(\mathcal{G},2)}_{\bowtie} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \end{pmatrix}$$

The equivalence relation \bowtie is balanced if and only if for each arrow specific 5×2 matrix rows 1, 3, and 5 are equal, and rows 2 and 4 are equal. However, this does not hold. Thus the equivalence relation $\bowtie = (135)(24)$ is not balanced.

On the other hand, let $\bowtie = (124)(3)(5)$. There are three equivalence classes $\overline{C}_1 = \{1, 2, 4\}$, $\overline{C}_2 = \{3\}$ and $\overline{C}_3 = \{5\}$. We generate new 5×3 matrices by adding vectors in columns 1, 2, 4:

$$\widetilde{M}_{(\mathcal{G},1)}_{\bowtie} = \begin{pmatrix} 2 & 0 & 0 \\ 2 & 0 & 0 \\ 0 & 1 & 1 \\ 2 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad \widetilde{M}_{(\mathcal{G},2)}_{\bowtie} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

The equivalence relation \bowtie is balanced if and only if for each arrow specific 5×2 matrix rows 1, 2, and 4 are equal. This is satisfied. Thus the equivalence relation $\bowtie = (124)(3)(5)$ is balanced. As a result, the quotient network \mathcal{G}/\bowtie corresponding to a balanced equivalence relation $\bowtie = (124)(3)(5)$ and the associated 3×3 arrow type specific adjacency matrices are given in Figure 4.4.

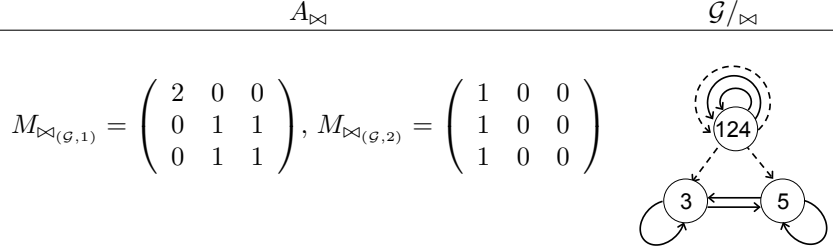


FIG. 4.4. The quotient network \mathcal{G}/\bowtie corresponding to a balanced equivalence relation $\bowtie = (124)(3)(5)$ and the associated 3×3 arrow type specific adjacency matrices.

◇

4.2. Lattice of balanced equivalence relations. Using the above computer algorithm, we can determine all balanced equivalence relations and corresponding quotient networks for a given coupled cell network. Now, using the refinement relation, we construct a complete lattice of balanced equivalence relations for a given coupled cell network.

Let p be the total number of balanced equivalence relations of a given coupled cell network. We aim to compute a $p \times p$ adjacency matrix $L = (l_{ij})$ for the lattice with entries 1 where \bowtie_i is covered by \bowtie_j , and 0 otherwise.

Step 1: Without loss of generality, order the p balanced equivalence relations by increasing rank (number of equivalence classes). This ensures that the top element is first and the bottom element last, and that the matrix L will be lower triangular.

Step 2: Construct $p \times p$ matrix $B = (b_{ij})$ with entries 1 where $\bowtie_i \prec \bowtie_j$ (\bowtie_i refines \bowtie_j) and 0 otherwise. This is almost the desired adjacency matrix, but it includes extra edges since refinement is not as strict as covering.

Step 3: Calculate $p \times p$ matrix $T = (t_{ij}) = B^2$. Non-zero entries t_{ij} indicate nodes i and j are connected by a path of length two via some intermediate third node k , thus $\bowtie_i \prec \bowtie_k \prec \bowtie_j$, meaning \bowtie_i is not covered by \bowtie_j . We can assume k is distinct from i and j since the diagonal entries of B are zero.

Step 4: Construct $p \times p$ matrix $L = (l_{ij})$ using $l_{ij} = 1$ if $b_{ij} = 1$ and $t_{ij} = 0$, and $l_{ij} = 0$ otherwise.

For larger lattices computing the full matrix T in step 3 is increasingly time consuming, and only a fraction of the values are needed in step 4. For computational efficiency, only where $b_{ij} = 1$ do we need to check if $t_{ij} = 0$. We do this by considering the existence of a two step path between lattice nodes i and j via node k (i.e. $b_{ik} = 1$ and $b_{kj} = 1$), and as a further optimization only those nodes k with $\text{rank}(i) < \text{rank}(k) < \text{rank}(j)$ need be considered.

The matrix L is the adjacency matrix for the p -node lattice, and defines the set of edges. Lattices are by convention drawn as diagrams with an up/down orientation with the top lattice element higher than the bottom lattice element. Additionally we require lattices nodes of the same rank (number of equivalence classes) to be shown at the same height.

EXAMPLE 4.3. Consider the five-cell homogeneous network \mathcal{G}_5 in Table 2.1. This

network has 5 balanced equivalence relations as shown in Figure 4.5, in rank order.

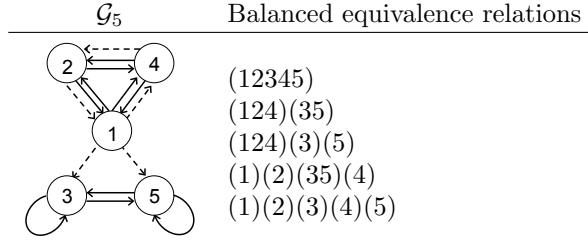


FIG. 4.5. Five-cell homogeneous network \mathcal{G}_5 in Table 2.1 and all possible 5 balanced equivalence relations.

We construct the 5×5 matrix B , which represents refinement relations between the 5 balanced equivalence relations. Then $T = B^2$ is a simple matrix multiplication, which is used to remove unwanted edges from matrix B to give L . Table 4.1 shows these three matrices. Matrix L is used as the adjacency matrix when drawing the lattice, with vertical positions dictated by the lattice node ranks (Figure 4.6, right).

B	$T = B^2$	L
$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 3 & 2 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$

TABLE 4.1

Three 5×5 matrices, B , T , and L defined in the lattice algorithm for the network \mathcal{G}_5 in Table 2.1.

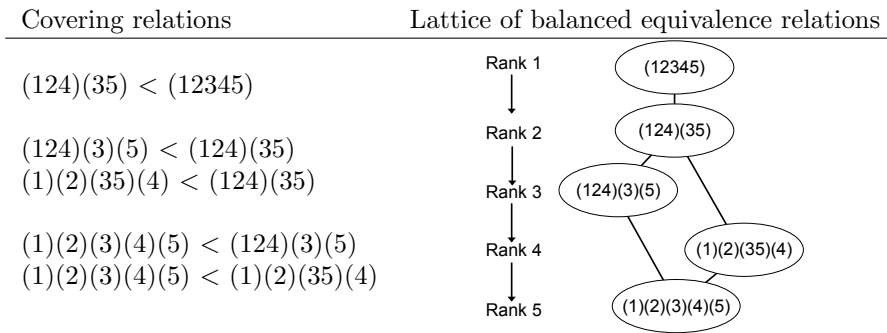


FIG. 4.6. The covering relations and the lattice of balanced equivalence relations of the five-cell homogeneous network \mathcal{G}_5 in Table 2.1.

◇

Computing all the balanced equivalence relations of a network size n scales with the number of equivalence relations, given by the Bell number, and is thus combina-

torial with the network size n . On a recent computer (2008 Apple Mac Pro) using a single CPU using the brute force approach with a single edge type, small networks take less than a second to compute, 10 nodes about 20 seconds, 11 nodes about 2 minutes, 12 nodes about 15 minutes, 13 nodes under 2 hours, and 14 nodes about 12 hours (with variation depending on the network topology).

We have also implemented the algorithm in [9], and generalized this to consider multiple arrow types. The result is equivalent to a single-phase simplification of the algorithm in [4], but less complicated to implement (see appendix). Where we have computed the full list of balanced coloring and identified the unique minimal balanced coloring, the results agree. As described above, this offers a shortcut when computing all the balanced equivalence relations, although for highly symmetric networks this optimization has limited benefit – and for regular networks offers no improvement. However, the time saving can be dramatic especially for random networks. As an example, a bidirectionally coupled chain (where the end nodes do not have self coupling) of up to 20 nodes takes under a second, 30 nodes takes about 20 seconds, and 40 nodes about 10 minutes.

The time (and the memory requirements) needed to compute the lattice scales quadratically with the number of lattice nodes. In the worst case of a fully connected network all equivalence relations are balanced, giving the largest possible lattice and the longest compute time, taking about a second for the 877 node lattice ($n = 7$), 20 seconds for the 4140 node lattice ($n = 8$), and 10 minutes for the 21147 node lattice ($n = 9$).

In short, in our current algorithm implementation computing the balanced colorings of *regular* networks more than 15 nodes is impractical, although inhomogeneous networks are much easier to deal with. Additionally, the computations could in principle be run in parallel across multiple CPU cores, giving a potential linear speed up.

5. Examples. The lattice of partial synchronies computed by the algorithm shown tells us about the existence of all possible partial synchronies determined by the given network structure. In this section, we select example topics from synchronized chaos and coupled neuron models, and demonstrate how a symbolic adjacency matrix can be defined for each example problem, and construct a complete lattice of all possible partial synchronies derived from the given networks structure. Some dynamical properties such as the stability of possible partial synchronies depends on the specific form of the given vector field. We demonstrate the numerical analysis of the stability of partial synchronies for the topic of synchronized chaos.

5.1. Coupled identical Rössler systems. We consider a bidirectional ring of six diffusively coupled Rössler systems $\mathbf{u}_i = (x_i, y_i, z_i) \in \mathbb{R}^3$ for $i = 1, 2, \dots, 6$:

$$\begin{aligned}\dot{x}_i &= -(y_i + z_i) + \epsilon(x_{i-1} - 2x_i + x_{i+1}), \\ \dot{y}_i &= x_i + ay_i, \\ \dot{z}_i &= b + (x_i - c)z_i,\end{aligned}$$

with periodic boundary conditions $x_0 = x_6$ and $x_7 = x_1$.

Since this is a regular network, the adjacency matrix consists of non-negative integers and the admissible vector field is defined by a single map \mathbf{f} , which is realized by the above defined system. Table 5.1 shows the adjacency matrix and the associated coupled cell system. The complete lattice of balanced equivalence relations of this network is given in Figure 5.1.

adjacency matrix	coupled cell system
$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$	$\begin{aligned} \dot{\mathbf{u}}_1 &= \mathbf{f}(\mathbf{u}_1, \overline{\mathbf{u}}_6, \overline{\mathbf{u}}_2) \\ \dot{\mathbf{u}}_2 &= \mathbf{f}(\mathbf{u}_2, \overline{\mathbf{u}}_1, \overline{\mathbf{u}}_3) \\ \dot{\mathbf{u}}_3 &= \mathbf{f}(\mathbf{u}_3, \overline{\mathbf{u}}_2, \overline{\mathbf{u}}_4) \\ \dot{\mathbf{u}}_4 &= \mathbf{f}(\mathbf{u}_4, \overline{\mathbf{u}}_3, \overline{\mathbf{u}}_5) \\ \dot{\mathbf{u}}_5 &= \mathbf{f}(\mathbf{u}_5, \overline{\mathbf{u}}_4, \overline{\mathbf{u}}_6) \\ \dot{\mathbf{u}}_6 &= \mathbf{f}(\mathbf{u}_6, \overline{\mathbf{u}}_5, \overline{\mathbf{u}}_1) \end{aligned}$

TABLE 5.1

Adjacency matrix which represents interactions among Rössler systems and the associated coupled cell system. The overline indicates that influence from coupling cells to that cell are identical, i.e., $f(x_i, \overline{x_j}, x_k)$ means $f(x_i, x_j, x_k) = f(x_i, x_k, x_j)$.

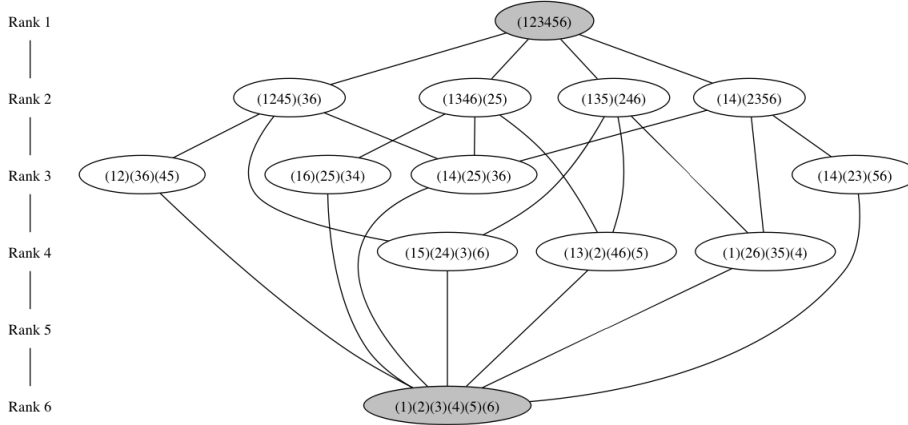


FIG. 5.1. Lattice of balanced equivalence relations of bidirectional ring of six diffusively coupled Rössler systems. Some partial synchronies are symmetrically related. For example the balanced equivalence relations (1245)(36), (1346)(25), and (14)(2356) are permutation symmetric and they give the same pattern of partial synchrony $(\alpha, \alpha, \beta, \alpha, \alpha, \beta)$ in a bidirectional ring.

In the numerical analysis, we take parameter values $a = 0.2$, $b = 0.2$, $c = 5.7$, and vary the coupling parameter ϵ for the stability analysis of synchrony subspaces of this specific vector field. The full synchrony subspace $\Delta = \{\mathbf{u}_1 = \dots = \mathbf{u}_6\}$ is globally stable from $\epsilon \approx 0.2$ to $\epsilon \approx 1.0$, and attracts all trajectories starting from randomly chosen initial conditions. With the loss of stability of full synchronization above $\epsilon \approx 1.0$, the synchrony subspace $\Delta_{\bowtie} = \{\mathbf{u}_1 = \mathbf{u}_3 = \mathbf{u}_5, \mathbf{u}_2 = \mathbf{u}_4 = \mathbf{u}_6\}$ becomes globally stable, and attracts all trajectories. Figure 5.2 (a) shows the behaviour of $x_1 - x_2$, which is the difference between the first internal variables of the Rössler systems \mathbf{u}_1 and \mathbf{u}_2 , when changing the parameter ϵ . This figure shows the gain and loss of stability of the full synchrony subspace. Figure 5.2 (b) shows the behaviour of $x_1 - x_3$, where the corresponding variables $\mathbf{u}_1, \mathbf{u}_3$ remain synchronized above $\epsilon \approx 1.0$. Only the behaviors of $x_1 - x_2$ and $x_1 - x_3$ are illustrated in Figure 5.2, qualitatively the same behaviors are observed for the other pairwise comparisons.

Rössler systems with x -component coupling, as shown in this example, are known to exhibit a phenomena called short wavelength bifurcations [20] in which the synchronous chaotic state loses its stability with an increase of coupling strength. The

desynchronization behaviour of three diffusively coupled Rössler systems with Neumann boundary conditions (a bidirectional chain with self-coupling of the end cells) was analysed in [11]. They showed that the singularity of the individual Rössler systems and the use of x -coupling impose the existence of equilibria that lie outside of the fully synchronous subspace for any coupling strength, and proposed a direct link to the mechanism of desynchronization.

A lattice-like hierarchy of synchrony subspaces of diffusively coupled identical systems was also discussed in [11]. For the chain network associated with Neumann boundary conditions, they hypothesized a clustering type hierarchy structure based on the number of nodes n and its divisors, which we have verified for up to $n = 15$ (see supplementary material). A related result for a linear chain with feedback in [33] gives an explicit lattice construction.

5.2. Coupled Lorenz systems with heterogeneous coupling. In the next example, we demonstrate the symbolic adjacency matrix can be interpreted not only as a network structure, but also as different coupling strengths of identical individual systems. Consider cluster synchronization in an ensemble of five globally coupled Lorenz systems $\mathbf{u}_i = (x_i, y_i, z_i) \in \mathbf{R}^3$ for $i = 1, \dots, 5$ with heterogeneous coupling:

$$\begin{aligned}\dot{x}_i &= \sigma(y_i - x_i) + \frac{1}{N_i} \sum_{j=1}^5 g_{ij}(x_i - x_j) \\ \dot{y}_i &= x_i(\rho - z_i) - y_i \\ \dot{z}_i &= x_i y_i - \beta z_i\end{aligned}$$

where $\sigma = 10$, $\rho = 28$, $\beta = 8/3$, $N_i = \sum_{j=1}^5 g_{ij}$, and the coupling matrix $G = (g_{ij})$ is defined as

$$G = \begin{pmatrix} 0 & 1 & 1 & 2 & 4 \\ 1 & 0 & 1 & 2 & 4 \\ 1 & 1 & 0 & 1 & 5 \\ 1 & 2 & 3 & 0 & 2 \\ 3 & 2 & 1 & 2 & 0 \end{pmatrix}$$

We may regard an integer value g_{ij} to be a weight from Lorenz system j to Lorenz system i , and different non-zero integer values to be different weights from the corresponding systems. Note that $N_i = 8$ for all $i = 1, \dots, 5$. Then symbolically, the above coupling matrix G can be represented with five different symbols, a , b , c , d , and e . Table 5.2 shows the symbolic adjacency matrix and the associated coupled cell system.

Figure 5.3 shows five globally coupled Lorenz systems with different weighted arrows represented in different colors and the associated lattice of balanced equivalence relations of the network. If all weights are identical, the corresponding lattice of partial synchronies is the same as the partition lattice of 5 elements with the Bell number $B_5 = 52$ lattice points. However, in this example with non-identical weights, there is only one non-trivial balanced equivalence relation given by (12)(3)(4)(5), which is found to be unstable by numerical analysis.

We note the intriguing phenomenon termed *bubbling* [6] is related to the stability of synchrony subspaces. When the dynamics on the synchrony subspace is a chaotic attractor, small perturbations along the transverse direction of the synchrony subspace

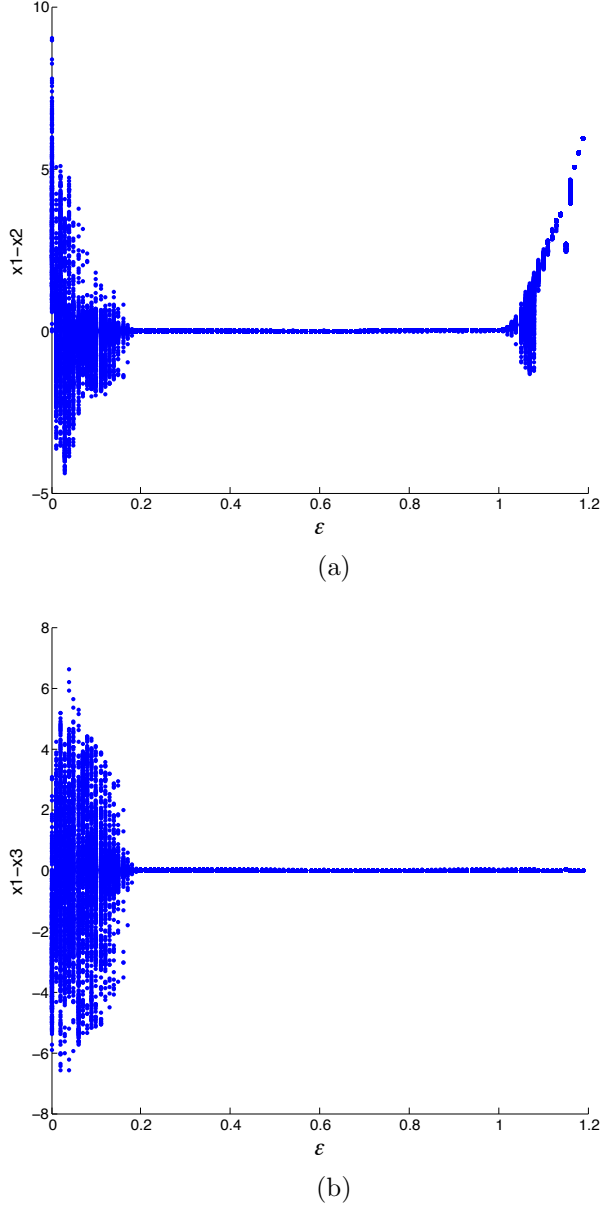


FIG. 5.2. The stability analysis of synchrony subspaces of a ring of six diffusively coupled Rössler systems. (a) The behaviour of $x_1 - x_2$ when changing the coupling parameter ϵ . This difference is chosen as a representative difference of variables associated with the full synchrony subspace $\Delta = \{\mathbf{u}_1 = \dots = \mathbf{u}_6\}$. The fully synchronous subspace Δ is stable between $\epsilon \approx 0.2$ and $\epsilon \approx 1.0$. (b) The behaviour of $x_1 - x_3$ when changing the coupling parameter ϵ , chosen as a representative difference of the variables associated with the synchrony subspace $\Delta_{\bowtie} = \{\mathbf{u}_1 = \mathbf{u}_3 = \mathbf{u}_5, \mathbf{u}_2 = \mathbf{u}_4 = \mathbf{u}_6\}$. This partial synchrony subspace is observed above $\epsilon \approx 1.0$ (when the fully synchrony subspace loses stability). The other parameter values are $a = 0.2$, $b = 0.2$, $c = 5.7$. For each fixed parameter value ϵ (in steps of 0.01 for $0 \leq \epsilon \leq 1.2$), 500 different initial conditions are generated and the difference $x_1 - x_2$ or $x_1 - x_3$ is plotted using the state variables after 40000 iterates with the time step $h = 0.01$.

symbolic adjacency matrix	coupled cell system
$G = \begin{pmatrix} 0 & a & a & b & d \\ a & 0 & a & b & d \\ a & a & 0 & a & e \\ a & b & c & 0 & b \\ c & b & a & b & 0 \end{pmatrix}$	$\begin{aligned} \dot{\mathbf{u}}_1 &= \mathbf{f}(\mathbf{u}_1, \overline{\mathbf{u}}_2, \overline{\mathbf{u}}_3, \mathbf{u}_4, \mathbf{u}_5) \\ \dot{\mathbf{u}}_2 &= \mathbf{f}(\mathbf{u}_2, \overline{\mathbf{u}}_1, \overline{\mathbf{u}}_3, \mathbf{u}_4, \mathbf{u}_5) \\ \dot{\mathbf{u}}_3 &= \mathbf{g}(\mathbf{u}_3, \overline{\mathbf{u}}_1, \overline{\mathbf{u}}_2, \overline{\mathbf{u}}_4, \mathbf{u}_5) \\ \dot{\mathbf{u}}_4 &= \mathbf{h}(\mathbf{u}_4, \mathbf{u}_1, \overline{\mathbf{u}}_2, \overline{\mathbf{u}}_5, \mathbf{u}_3) \\ \dot{\mathbf{u}}_5 &= \mathbf{h}(\mathbf{u}_5, \mathbf{u}_1, \overline{\mathbf{u}}_2, \overline{\mathbf{u}}_4, \mathbf{u}_3) \end{aligned}$

TABLE 5.2

Symbolic adjacency matrix for network in Figure 5.3 which represents different weights on arrows, and the associated coupled cell system.

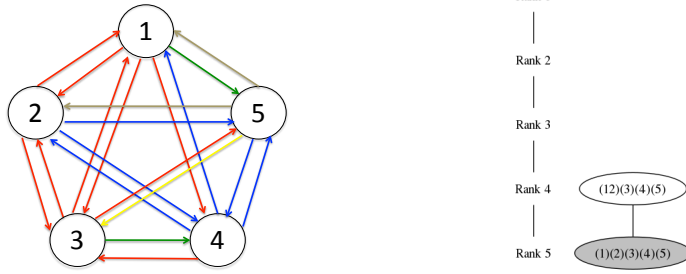


FIG. 5.3. Five globally coupled Lorenz systems and the associated lattice of balanced equivalence relations, which has a unique non-trivial balanced equivalence relation (12)(3)(4)(5).

can induce intermittent bursting for some systems. This bubbling phenomenon is observed in synchrony subspaces corresponding to balanced coloring (see the example system (14.1) in [18]).

5.3. Coupled neurons on a random network. The Aldis [4] or Belykh and Hasler [9] algorithm finds the minimal balanced coloring, which is the balanced equivalence relation with the minimal number of colors (i.e. the minimal number of the synchronized clusters), and thus the top lattice node. Belykh and Hasler demonstrated this using a coupled identical Hindmarsh-Rose model [21] with 30 neurons generated by randomly choosing bidirectional identical couplings between any two nodes with a small probability. Even though this network has only one type of cell and one type of coupling, it is not regular. This network has no apparent symmetry using the circular layout (Figure 5.4(a)) which hides a local reflectional symmetry (Figure 5.4(b)). The minimal balanced coloring has 23 colors (i.e. 23 synchronized clusters). Our algorithm shows the lattice of balanced equivalence relations contains only two lattice points, the trivial bottom lattice node (all distinct) and this one non-trivial balanced coloring. Modifying this network by adding random edges or rewiring existing edges can generate a more complex lattice, for example ten lattice nodes with a minimal balanced coloring of 18 clusters (Figure 5.4(c)). The Python code in the supplementary materials finds the lattices for both of these networks. A systematic exploration of the lattices of random networks, such as those generated by rewiring or the Watts-Strogatz model [40], is an area for future work.

5.4. Excitatory/Inhibitory coupled neurons. The network of inhibitory coupled FitzHugh-Nagumo model neurons shown in Figure 5.5(a) is discussed in

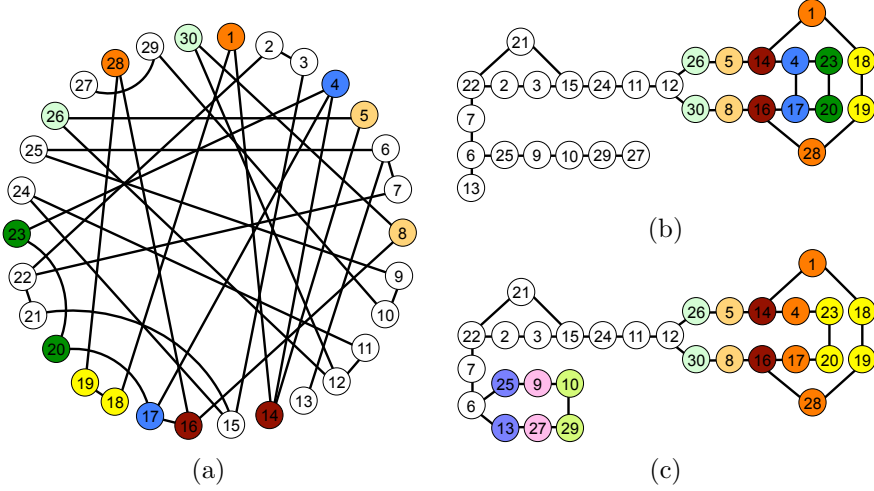


FIG. 5.4. Thirty neuron networks with bidirectional coupling. Colored nodes represent clusters in the minimal balanced coloring, with uncolored (white) nodes for distinct uni-clusters. Here (a) and (b) show the same network, from Figure 3(b) in [9], with 23 clusters. The final figure (c) shows a rewired network connecting nodes 13 and 27 in place of 4 and 17, resulting in a richer lattice of ten nodes, with 18 clusters in the minimal balanced coloring. Removing the coupling between 4 and 17 increases the local symmetry, as does coupling nodes 13 and 27.

[30, 31]. Table 5.3 shows the the 9×9 adjacency matrix $C = (c_{ij})$ with integer entries and the associated coupled cell system. A very similar neural network topology (deleting the arrow from neuron 1 to 5) is studied in [13] using a discrete map instead of ODEs. We remark that our algorithm shows both network structures have the same lattice of 27 balanced equivalence relations (see lattice generated by the Python code in the supplementary materials). The top lattice node is the synchronized cluster pattern $(19)(2378)(46)(5)$ which was discussed in [13].

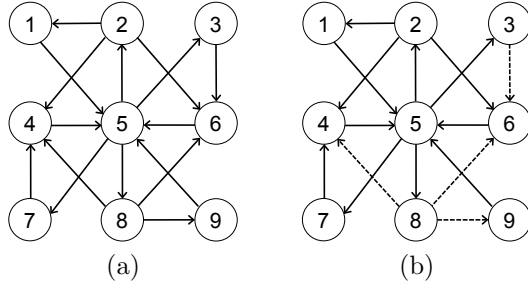


FIG. 5.5. Nine neurons connected with (a) one coupling type as in [30, 31], (b) two coupling types (excitatory and inhibitory). Solid arrows represent inhibitory coupling, dashed lines excitatory. In (b) the arrows from nodes 3 and 8 are excitatory.

This network structure was studied as a *winnerless competition network* [30, 31], in which cluster states (unstable saddle states) are connected along a heteroclinic orbit. Such cluster states can correspond to balanced polydiagonals (invariant subspaces), which are defined by balanced equivalence relations [8, 7]. Thus the lattice of balanced equivalence relations might potentially be used to elucidate the possible robust heteroclinic cycles.

adjacency matrix	coupled cell system
$C = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$	$\begin{aligned} \dot{\mathbf{u}}_1 &= \mathbf{f}(\mathbf{u}_1, \mathbf{u}_2) \\ \dot{\mathbf{u}}_2 &= \mathbf{f}(\mathbf{u}_2, \mathbf{u}_5) \\ \dot{\mathbf{u}}_3 &= \mathbf{f}(\mathbf{u}_3, \mathbf{u}_5) \\ \dot{\mathbf{u}}_4 &= \mathbf{g}(\mathbf{u}_4, \overline{\mathbf{u}_2, \mathbf{u}_7, \mathbf{u}_8}) \\ \dot{\mathbf{u}}_5 &= \mathbf{h}(\mathbf{u}_5, \overline{\mathbf{u}_1, \mathbf{u}_4, \mathbf{u}_6, \mathbf{u}_9}) \\ \dot{\mathbf{u}}_6 &= \mathbf{g}(\mathbf{u}_6, \overline{\mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_8}) \\ \dot{\mathbf{u}}_7 &= \mathbf{f}(\mathbf{u}_7, \mathbf{u}_5) \\ \dot{\mathbf{u}}_8 &= \mathbf{f}(\mathbf{u}_8, \mathbf{u}_5) \\ \dot{\mathbf{u}}_9 &= \mathbf{f}(\mathbf{u}_9, \mathbf{u}_8) \end{aligned}$

TABLE 5.3

Adjacency matrix which represents the network topology of 9 coupled neurons in Figure 5.5(a), and the associated coupled cell system. Note that this network is not regular since it has three input equivalence classes. Note also that the single coupling type constrains the linearized external couplings of the three maps \mathbf{f} , \mathbf{g} and \mathbf{h} to be the same.

To demonstrate multiple arrow types, we modified the previous example to consider two coupling types, excitatory or inhibitory, as in Figure 5.5(b). By changing the outputs of neurons 3 and 8 to be excitatory (i.e. changing four couplings), the number of balanced equivalence relations decreased from 27 to 15 (see lattices generated by the Python code in the supplementary materials). Table 5.4 shows the corresponding symbolic adjacency matrix and the associated coupled cell system.

symbolic adjacency matrix	coupled cell system
$C = \begin{pmatrix} 0 & a & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a & 0 & 0 & 0 & 0 \\ 0 & a & 0 & 0 & 0 & 0 & a & b & 0 \\ a & 0 & 0 & a & 0 & a & 0 & 0 & a \\ 0 & a & b & 0 & 0 & 0 & 0 & b & 0 \\ 0 & 0 & 0 & 0 & a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & b & 0 \end{pmatrix}$	$\begin{aligned} \dot{\mathbf{u}}_1 &= \mathbf{f}(\mathbf{u}_1, \mathbf{u}_2) \\ \dot{\mathbf{u}}_2 &= \mathbf{f}(\mathbf{u}_2, \mathbf{u}_5) \\ \dot{\mathbf{u}}_3 &= \mathbf{f}(\mathbf{u}_3, \mathbf{u}_5) \\ \dot{\mathbf{u}}_4 &= \mathbf{k}(\mathbf{u}_4, \mathbf{u}_8, \overline{\mathbf{u}_2, \mathbf{u}_7}) \\ \dot{\mathbf{u}}_5 &= \mathbf{l}(\mathbf{u}_5, \overline{\mathbf{u}_1, \mathbf{u}_4, \mathbf{u}_6, \mathbf{u}_9}) \\ \dot{\mathbf{u}}_6 &= \mathbf{m}(\mathbf{u}_6, \mathbf{u}_2, \overline{\mathbf{u}_3, \mathbf{u}_8}) \\ \dot{\mathbf{u}}_7 &= \mathbf{f}(\mathbf{u}_7, \mathbf{u}_5) \\ \dot{\mathbf{u}}_8 &= \mathbf{f}(\mathbf{u}_8, \mathbf{u}_5) \\ \dot{\mathbf{u}}_9 &= \mathbf{n}(\mathbf{u}_9, \mathbf{u}_8) \end{aligned}$

TABLE 5.4

Symbolic adjacency matrix which represents the network topology of 9 coupled neurons with excitatory (symbol 'a') and inhibitory (symbol 'b') coupling in Figure 5.5(b), and the associated coupled cell system.

6. Conclusions. Networks in real world applications are inhomogeneous. Individual systems in a network play different roles and they interact with each other in various ways. For example even in a simplified representation of gene regulatory networks, genes or proteins can interact either by activation or inhibition. We encoded different types of interaction in such inhomogeneous networks using a symbolic adjacency matrix. We considered possible partial synchronies of a given network, where the network elements can be grouped into clusters whose dynamics are self-

synchronous. We are particularly interested in partial synchronies which are solely determined by the structure (topology) of the network, rather than the specific dynamics (such as function forms and parameter values). Such robust patterns of synchrony are associated with balanced equivalence relations, which can be determined by a matrix computation on the symbolic adjacency matrix. These symbolic adjacency matrices can alternatively be expressed as a linear combination of integer entry adjacency matrices for each coupling type, and the matrix computation applied to each arrow type matrix individually (as in the provided Python program). The later is simpler to implement as most programming languages or software packages do not support symbolic matrices. Using the Symbolic Python library (<http://www.sympy.org/>) our example program can be modified to work on symbolic matrices (not shown as the extra dependency complicates installation for no practical benefit).

The symbolic adjacency matrix therefore specifies the set of balanced equivalence relations for a network. From these the refinement relation gives a complete lattice. Rather than obtaining the lattice in this way by exhaustive computation, for the special case of regular networks with simple eigenvalues, Kamei [22, 23, 24] showed how to construct the lattice from building blocks related to the eigenvector/eigenvalues of the adjacency matrix, and use it to predict the existence of codimension-one steady-state bifurcation branches from the fully synchronous state, and to classify synchrony-breaking bifurcation behaviors. Results in [16, 28, 1, 3, 17] also relate the eigenvalues of the Jacobian of a coupled cell system with the eigenvalues of the adjacency matrix of a homogeneous network for synchrony-breaking bifurcation analysis. The connections between the algebraic properties of the lattice and the network dynamics are potentially of wide interest.

In general however, such theoretical approaches for the explicit construction of the lattice do not yet exist, leaving the “brute force” approach of calculating all the possible balanced equivalence relations as the only currently viable route. We hope that use of this algorithm will facilitate further theoretical work, and stimulate investigation linking lattice properties and synchronous dynamics, and ultimately links between network structure and dynamics.

Acknowledgments. H. Kamei thanks Prof. Ian Stewart for his supervision while the foundations of this work was carried out at the University of Warwick.

References.

- [1] M. A. D. AGUIAR, A. P. DIAS, M. GOLUBITSKY, AND M. C. A. LEITE, *Homogeneous coupled cell networks with S_3 -symmetric quotient.*, Discrete and Continuous Dynam. Sys. Supplement, (2007), pp. 1–9.
- [2] M. A. D. AGUIAR AND A. P. S. DIAS, *Minimal coupled cell networks.*, Nonlinearity, 20 (2007), pp. 193–219.
- [3] M. A. D. AGUIAR, A. P. S. DIAS, M. GOLUBITSKY, AND M. C. A. LEITE, *Bifurcation from regular quotient networks: A first insight.*, Phys. D, 238 (2009), pp. 137–155.
- [4] J. W. ALDIS, *A polynomial time algorithm to determine maximal balanced equivalence relations.*, Int. J. Bifurcation and Chaos, 18 (2008), pp. 407–427.
- [5] A. ARENAS, A. DIAZ-GUILERA, J. KURTHS, Y. MORENO, AND C. ZHOU, *Synchronization in complex networks.*, Phys. Rep., 469 (2008), pp. 93–153.
- [6] P. ASHWIN, J. BUESCU, AND I. STEWART, *Bubbling of attractors and synchronisation of chaotic oscillators.*, Phys. Lett. A, 193 (1994), pp. 126–139.

- [7] P. ASHWIN, O. KARABACAK, AND T. NOWOTNY, *Criteria for robustness of heteroclinic cycles in neural microcircuits.*, J. Math. Neurosci., 1 (2011), pp. 1–18.
- [8] P. ASHWIN, G. OROSZ, J. WORDSWORTH, AND S. TOWNLEY, *Dynamics on networks of cluster states for globally coupled phase oscillators.*, SIAM J. Appl. Dyn. Syst., 6 (2007), pp. 728–758.
- [9] I. BELYKH AND M. HASLER, *Mesoscale and clusters of synchrony in networks of bursting neurons.*, Chaos, 21 (2011). 016106.
- [10] V. BELYKH, G. OSIPOV, V. PETROV, J. SUYKENS, AND J. VANDEWALLE, *Cluster synchronization in oscillatory networks.*, Chaos, 18 (2008). 037106.
- [11] V. BELYKH, I. BELYKH, AND M. HASLER, *Hierarchy and stability of partially synchronous oscillations of diffusively coupled dynamical systems.*, Phys. Rev. E, 62 (2000), pp. 6332–6345.
- [12] S. BOCCALETTI, V. LATORA, Y. MORENO, M. CHAVEZ, AND D. U. HWANG, *Complex networks: Structure and dynamics.*, Phys. Rep., 424 (2002), pp. 175–308.
- [13] J. M. CASADO, *Transient activation in a network of coupled map neurons.*, Phys. Rev. Lett., 91 (2003). 208102.
- [14] B. A. DAVEY AND H. A. PRIESTLEY, *Introduction to Lattices and Order.*, Cambridge University Press, Cambridge, 1990.
- [15] J. ELLSON, E. R. GANSNER, E. KOUTSOFIOS, S. C. NORTH, AND G. WOODHULL, *Graphviz - open source graph drawing tools*, in Graph Drawing, Springer-Verlag, 2001, pp. 483–484.
- [16] T. ELMHIRST AND M. GOLUBITSKY, *Nilpotent hopf bifurcations in coupled cell systems*, SIAM J. Appl. Dyn. Syst., 5 (2006), pp. 205–251.
- [17] M. GOLUBITSKY AND R. LAUTERBACH, *Bifurcations from synchrony in homogeneous networks: Linear theory.*, SIAM J. Appl. Dyn. Syst., 8 (2009), pp. 40–75.
- [18] M. GOLUBITSKY AND I. STEWART, *Nonlinear dynamics of networks: the groupoid formalism.*, Bull. Amer. Math. Soc., 43 (2006), pp. 305–364.
- [19] M. GOLUBITSKY, I. STEWART, AND A. TÖRÖK, *Patterns of synchrony in coupled cell networks with multiple arrows.*, SIAM J. Appl. Dyn. Syst., 4 (2005), pp. 78–100.
- [20] J. HEAGY, L. PECORA, AND T. CARROLL, *Short wavelength bifurcations and size instabilities in coupled oscillator systems.*, Phys. Rev. Lett., 74 (1995). 188101.
- [21] J. HINDMARSH AND M. ROSE, *A model of neuronal bursting using three coupled first order differential equations.*, Proc. R. Soc. Lond. Ser. B, 221 (1984), pp. 87–102.
- [22] H. KAMEI, *Interplay between network topology and synchrony-breaking bifurcation: Four-cell coupled cell networks.*, PhD Thesis, University of Warwick, (2008).
- [23] ———, *Construction of lattices of balanced equivalence relations for regular homogeneous networks using lattice generators and lattice indices.*, Int. J. Bifurcation and Chaos, 19 (2009), pp. 39691–3705.
- [24] ———, *The existence and classification of synchrony-breaking bifurcations in regular homogeneous networks using lattice structures.*, Int. J. Bifurcation and Chaos, 19 (2009), pp. 3707–3732.
- [25] A. KOSESKA, E. ULLNER, E. VOLKOV, J. KURTHS, AND J. GARCÍA-OJALVO, *Cooperative differentiation through clustering in multicellular populations.*, J. Theoret. Biol., 263 (2010), pp. 189–202.

- [26] Y. KURAMOTO AND D. BATTOGTOKH, *Coexistence of coherence and incoherence in nonlocally coupled phase oscillators*, Nonl. Phen. in Complex Systems, 5 (2002), pp. 380–385.
- [27] P. LANCASTER AND M. TISMENETSKY, *The Theory of Matrices Second Edition with Applications*, Academic Press, San Diego, 1985.
- [28] M. C. A. LEITE AND M. GOLUBITSKY, *Homogeneous three-cell networks.*, Nonlinearity, 19 (2006), pp. 2313–2363.
- [29] M. E. J. NEWMAN, *The structure and function of complex networks.*, SIAM Review, 45 (2003), pp. 167–256.
- [30] M. I. RABINOVICH, R. HUERTA, A. VOLKOVSKII, H. D. I. ABARBANEL, M. STOPFER, AND G. LAURENT, *Dynamical coding of sensory information with competitive networks.*, J. Physiol., 94 (2000), pp. 465–471.
- [31] M. I. RABINOVICH, A. VOLKOVSKII, R. HUERTA, H. D. I. ABARBANEL, AND G. LAURENT, *Dynamical encoding by networks of competing neuron groups: Winnerless competition.*, Phys. Rev. Lett., 87 (2001). 068102.
- [32] I. STEWART, *Self-organization in evolution: a mathematical perspective*, Philos. Transact. A. Math. Phys. Eng. Sci., 361 (2003), pp. 1101–1123.
- [33] ———, *The lattice of balanced equivalence relations of a coupled cell network.*, Math. Proc. Camb. Phil. Soc., 143 (2007), pp. 165–183.
- [34] I. STEWART, M. GOLUBITSKY, AND M. PIVATO, *Symmetry groupoids and patterns of synchrony in coupled cell networks*, SIAM J. Appl. Dyn. Syst., 2 (2003), pp. 609–646.
- [35] S. H. STROGATZ, *Exploring complex networks.*, Nature, 410 (2001), pp. 268–276.
- [36] W. T. TUTTE, *Graph Theory, Encyclopaedia of Mathematics and Its Applications, Vol. 21, ed. Rota, G.-C.*, Addison-Wesley, Menlo Park, 1984.
- [37] P. UHLHAAS AND W. SINGER, *Neural synchrony in brain disorders: Relevance for cognitive dysfunctions and pathophysiology.*, Neuron, 52 (2006), pp. 155–168.
- [38] J. WANG AND A. CHEN, *Partial synchronization in coupled chemical chaotic oscillators.*, J. Comput. Appl. Math., 233 (2010), pp. 1897–1904.
- [39] X. F. WANG, *Complex networks: Topology, dynamics and synchronization.*, Int. J. Bifurcation and Chaos, 12 (2002), pp. 885–916.
- [40] D. J. WATTS AND S. H. STROGATZ, *Collective dynamics of ‘small-world’ networks.*, Nature, 393 (1998), pp. 409–410.
- [41] R. WILSON, *Introduction to Graph Theory, 3rd edition*, Longman, Harlow, 1985.
- [42] Y. ZHANG, G. HU, H. CERDEIRA, S. CHEN, T. BRAUN, AND Y. YAO, *Partial synchronization and spontaneous spatial ordering in coupled chaotic systems.*, Phys. Rev. E, 63 (2001). 026211.

7. Appendix - Algorithm to find the top lattice node. The algorithm we use is a generalization of Belykh and Hasler [9] to consider multiple arrow types, or equivalently a simplification of the Aldis [4] algorithm where phase one is eliminated at the cost of counting absent arrow type/tail-node color combinations. This simplification makes it easier to implement than the original Aldis algorithm, yet it is still more than fast enough for our needs. The nine-cell network in Figure 5.5(b) with two arrow types is used as an example:

Step 0. Start by assigning the same color (node class) to each node, here shown in red. If multiple node types are considered as in Aldis, then each node type would be allocated a unique color. Aldis also classifies the arrows but we skip with that.

Step 1. In each step compute the “input driven refinement” by tallying the inputs to each node according to the color of the node the input is from, *and* the arrow type. After tabulation, unique input combinations give the next node partition.

See Figure 7.1. Here we have one node color (red), and two arrow types (solid and dashed), so for each node there are two input counts (solid from red, dashed from red). Here Aldis would also compute the same two input counts per node.

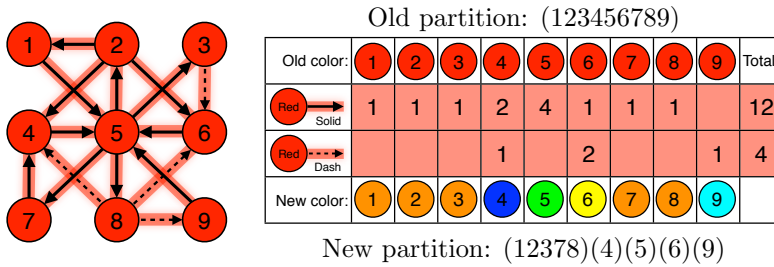


FIG. 7.1. Finding Top Lattice Node (Step 1), input driven refinement of partition (123456789).

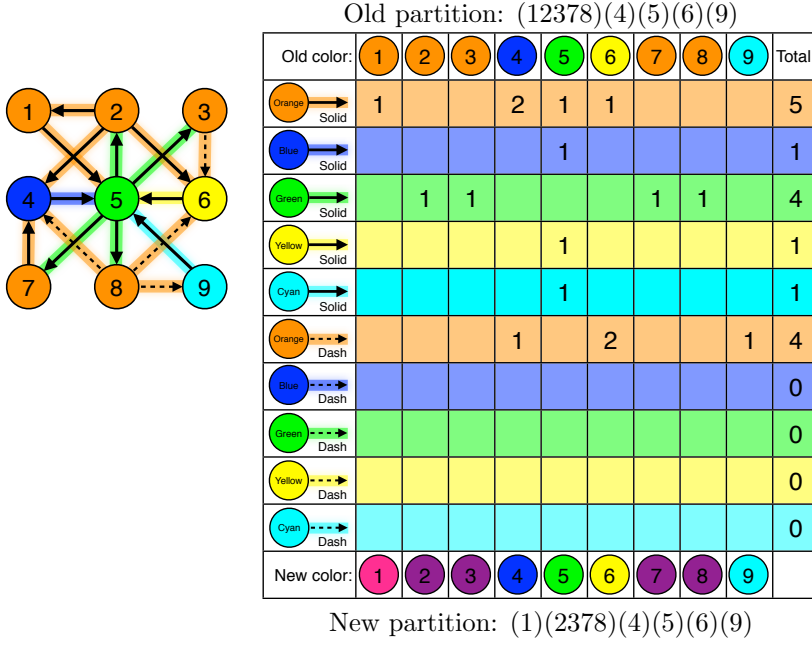
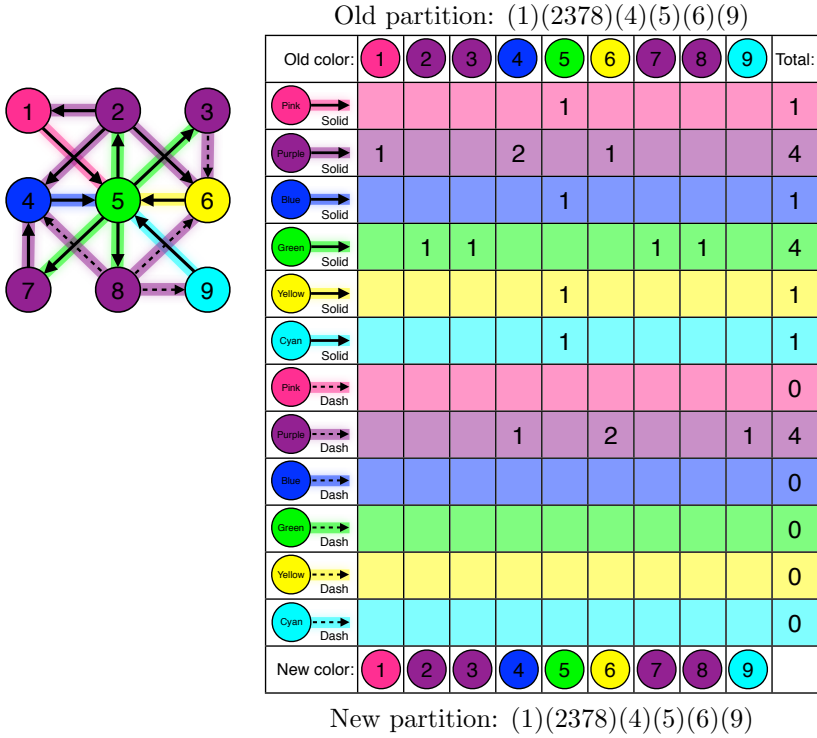
We observe five unique input combinations, and so assign them five colors as the “input driven refinement” of the (trivial) input partition. For instance, nodes with one solid input from a red node only have been assigned the new partition color orange.

Step 2. There are now five node colors (shown here as orange, blue, green, yellow and cyan). Thus with two arrow types (solid and dashed), we consider ten input types ($5 \times 2 = 10$; solid from orange, ..., solid from cyan, dashed from orange, ..., dashed from cyan). See Figure 7.2.

We observe six unique input combinations, giving six colors in the new node partition. The colors shown are arbitrary, and in the implementation are simply integers assigned incrementally. For this example we have reused the colors blue, green, yellow and cyan since those node groupings are unchanged. The former orange nodes have now been divided into pink and purple nodes.

Notice that of the ten possible input types tabulated here, the last four are absent. The Aldis algorithm avoids counting these.

Step 3. There are now six node colors (pink, purple, blue, green, yellow and cyan), so with two arrow types (solid and dashed) we consider twelve input types ($6 \times 2 = 12$; solid from pink, ..., solid from cyan, dashed from pink, ..., dashed from cyan). See Figure 7.3. At this iteration the partition of nodes is unchanged, and the algorithm halts. This gives the top lattice node.

FIG. 7.2. Finding Top Lattice Node (Step 2), refinement of partition $(12378)(4)(5)(6)(9)$.FIG. 7.3. Finding Top Lattice Node (Step 3), halts at partition $(1)(2378)(4)(5)(6)(9)$.

As in the previous step, some of the possible input types tabulated do not occur (five out of twelve), and the Aldis algorithm avoids counting these.

Remark. When comparing the tables in steps 2 and 3, reusing the same color for node groups preserved between iterations highlights that the blue, green, yellow and cyan rows are unchanged. In this example at step 3 only the new pink and purple rows need be calculated (replacing the orange rows in step 2). This suggests a possible speed optimization when finding the top lattice node.

Differences between Aldis (2008) and our implementation. In the above we have noted that the Aldis algorithm avoids computing the zero rows present in our tally tables. This is done by an additional phase in each iteration which tracks the arrow type and tail node color combinations as arrow equivalence classes, shown schematically in Figure 7.4. This figure shows ancestry trees of the node partitions (left) and the observed combinations of arrow type (solid or dashed) with tail node color (right). By tracking the observed arrow type and tail-node color combinations explicitly, absent potential combinations need not be counted (i.e. dashed arrows from blue, green, yellow, cyan or pink nodes).

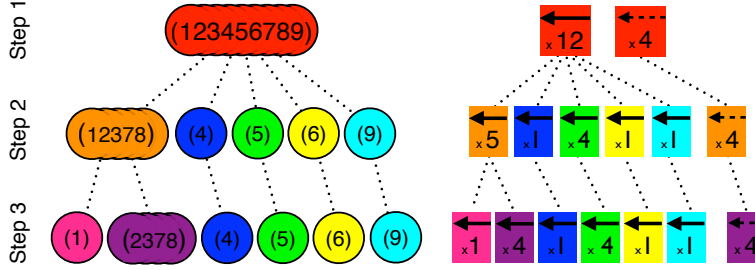


FIG. 7.4. Tree of node and arrow partitions as used in the Aldis algorithm. The node partitions on the left are (123456789) at step 1 (all red), (12378)(4)(5)(6)(9) at step two (five colors), and finally (1)(2378)(4)(5)(6)(9) at step 3 (six colors). The arrow partitions are shown in the disjoint tree on the right, using our notation combining the arrow type (solid or dashed) and tail-node partition color. The numbers in each box indicate the number of arrows of that type from nodes of that color, for example in step 1 there are 12 solid arrows from red nodes.

The solid arrow tree and the dashed arrow trees in Figure 7.4 (right) are both subtrees of the node partition tree (left). Our approach can be viewed as implicitly using the full node partition tree for each arrow type, at the cost of including redundant zero branches. This is a tradeoff between algorithmic complexity (Aldis) versus additional memory and computational overhead (not noticeable on the graph sizes considered). We expect the approach of Aldis to be most beneficial in large networks with many arrow types.